

华迈网络设备 SDK

使用手册

V2.2.0

说明

前言

感谢使用我公司产品，我们将竭诚为您服务。

本手册可能包含文字描述错误或其他错误，若文字描述与程序库头文件不一致，以头文件为准。

我公司将对前端设备中嵌入的程序定期更新，本手册内容可能会随之更新，恕不另行通知；更新内容会体现在本手册最新的版本中。

更新记录

2013-9-11 V1.0.1.8142

正式版本发布。

2014-5-6 V1.2.0.10982

- 1、连接模块重大升级。
- 2、内部架构优化。

2014-10-8 V1.2.0.13267

- 1、修复一些导致崩溃的和逻辑错误的 Bug。
- 2、增加了一些手机平台使用的接口。

2017-04-21 V2.2.0

- 1、SDK 版本更新至 2.2.0:
 - 1) 支持 H265 解码
 - 2) 其他问题修复和优化
- 2、增加接口 [hm_video_reset](#)，重置解码器，不释放，主要用于 SDK 断线重连重新等待第一个 I 帧才开始解码
- 3、接口 [hm_video_decode_yuv](#) 增加支持 H265 解码功能
- 4、文档完善

概览

自定义类型说明

	内置类型	自定义类型	说明
1	signed char	int8	
2	unsigned char	uint8	
3	signed short	int16	
4	unsigned short	uint16	
5	signed int	int32	
6	unsigned int	uint32	
7	__int64 / long long	int64	Windows 下使用 __int64 内置类型

8	unsigned __int64 / unsigned long long	uint64	同上
9	unsigned int	hm_result	返回值错误码
10	void*	window_handle	窗口句柄
11	void*	user_id	登录设备的返回值
12	void*	video_handle	实时视频句柄
13	void*	audio_handle	实时音频句柄
14	void*	talk_handle	对讲句柄
15	void*	alarm_handle	报警句柄
16	void*	record_handle	手动录像句柄
17	void*	find_file_handle	录像查找句柄
18	void*	playback_handle	录像回放句柄
19	void*	get_file_handle	录像下载句柄
20	void*	find_picture_handle	图片查找句柄
21	void*	get_picture_handle	图片下载句柄

22	void*	search_wifi_handle	WIFI 搜索句柄
23	void*	get_online_user_handle	获取在线用户句柄
24	void*	upgrade_handle	升级句柄
25	void*	server_id	平台操作句柄
26	void*	tree_handle	设备树句柄
27	void*	node_handle	设备句柄
28	void*	audio_codec_handle	音频编解码句柄
29	void*	video_codec_handle	视频解码句柄
30	void*	yuv_handle	yuv 对象句柄
31	void*	bitmap_handle	rgb 对象句柄
32	void*	audio_capture_handle	音频采集句柄
33	void*	audio_player_handle	音频播放器句柄
34	void*	local_record_handle	本地录像句柄
35	void*	local_playback_handle	本地回放句柄

36	void*	port_handle	端口句柄
37	void*	listen_handle	监听句柄
38	void*	user_data	用户自定义数据
39	float	real32	
40	double	real64	

错误码说明

	错误枚举	错误码值	说明
1	HMEC_OK	0	成功
2	HMEC_GEN_SDK_INIT	0x01000001	初始化失败
3	HMEC_GEN_CREATE_HANDLE	0x01000002	创建句柄失败
4	HMEC_GEN_PARAM_INVALID	0x01000003	参数无效
5	HMEC_GEN_NODE_INVALID	0x01000004	节点无效
6	HMEC_GEN_URL_INVALID	0x01000005	URL 无效

7	HMEC_GEN_FIND_FILE_OVER	0x01000006	文件查找 完毕
8	HMEC_GEN_NOT_SUPPORT	0x01000007	接口不被 支持
9	HMEC_ERR_PU_UNDEF	0x00800001	未定义错 误
10	HMEC_ERR_PU_BUSY	0x00800002	设备忙
11	HMEC_ERR_PU_BAD_PARAM	0x00800003	错误参数
12	HMEC_ERR_PU_BAD_FORMAT	0x00800004	错误格式
13	HMEC_ERR_PU_INTERNAL	0x00800005	内部错误
14	HMEC_ERR_PU_UNREG_CMD	0x00800006	不能识别 的命令
15	HMEC_ERR_PU_MAX_CONNECTION	0x00800007	最大连接 数限制
16	HMEC_ERR_PU_NOT_LOGIN	0x00800008	未认证
17	HMEC_ERR_PU_BAD_USER	0x00800009	错误的用 户名
18	HMEC_ERR_PU_UNSUPPORTED	0x0080000A	不支持的 参数
19	HMEC_ERR_PU_DISK_FULL	0x0080000B	磁盘满
20	HMEC_ERR_PU_DISK_NOT_EXIST	0x0080000C	存储介质 不存在

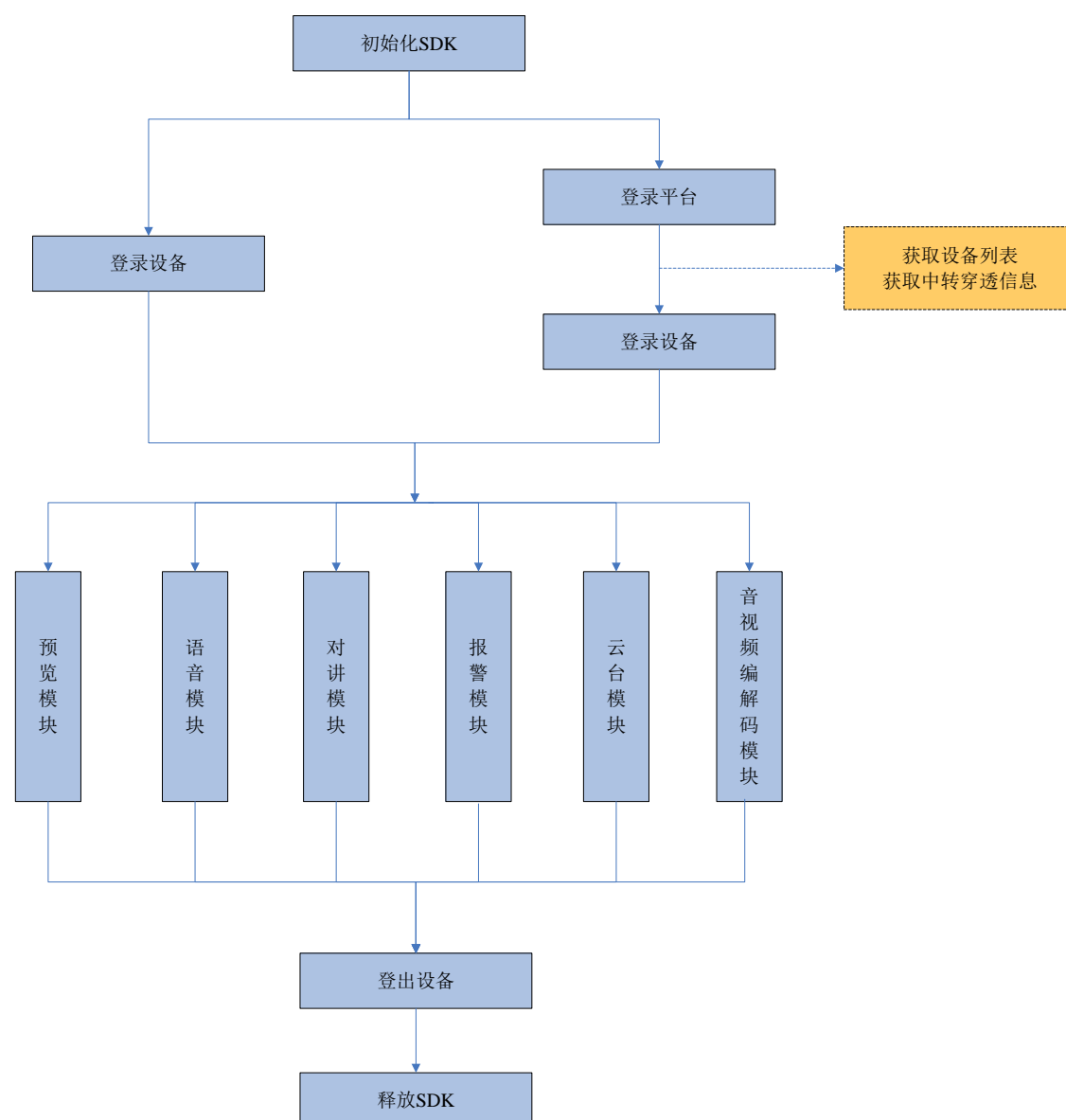
21	HMEC_ERR_PU_CHANNEL_USED	0x0080000D	通道忙
22	HMEC_ERR_PU_CHANNEL_CLOSED	0x0080000E	通道已关闭
23	HMEC_ERR_PU_CHANNEL_ERR	0x0080000F	错误的通道号
24	HMEC_ERR_PU_OPERATION_ABORT	0x00800050	操作中断
25	HMEC_ERR_PU_READ_LESS_LENGTH	0x00800051	数据长度太小
26	HMEC_ERR_PU_NONSUPPORT	0x00800052	不支持的操作
27	HMEC_ERR_PU_SEND_DATA_INVALID	0x00800053	发送数据无效
28	HMEC_ERR_SERVER_NETWORK	0x00700001	网络错误
29	HMEC_ERR_SERVER_COMMAND	0x00700002	命令错误
30	HMEC_ERR_SERVER_USER_OR_PWD	0x00700003	用户名或密码错误
31	HMEC_ERR_SERVER_NOT_AUTH	0x00700004	未认证
32	HMEC_ERR_SERVER_ALREADY_LOGIN	0x00700005	重复登录
33	HMEC_ERR_SERVER_GET_GROUP_LIST	0x00700006	获取分组失败
34	HMEC_ERR_SERVER_GET_DEVICE_LIST	0x00700007	获取设备失败

35	HMEC_ERR_SERVER_GET_USER_INFO	0x00700008	获取用户 信息失败
36	HMEC_ERR_SERVER_GET_TRANSFER_INFO	0x00700009	获取中 转失败
37	HMEC_ERR_SERVER_BIND_DEVICE	0x0070000A	绑定设备 失败
38	HMEC_ERR_SERVER_DELETE_GROUP	0x0070000B	删除设备 失败
39	HMEC_ERR_SERVER_MOVE_GROUP	0x0070000C	移动分组 失败
40	HMEC_ERR_SERVER_USER_CANCEL	0x0070000D	用户取消 操作
41	HMEC_ERR_SERVER_MAXERR	0x0070000E	未知错误
42	HMEC_ERR_VIDEO_FIND_DECODER	0x00300001	未知解码 类型
43	HMEC_ERR_VIDEO_ALLOC_CONTEXT	0x00300002	分配上下 文失败
44	HMEC_ERR_VIDEO_CODEC_OPEN	0x00300003	打开解码 器失败
45	HMEC_ERR_VIDEO_ALLOC_FRAME	0x00300004	分配存储 器失败
46	HMEC_ERR_VIDEO_STATE_DECODING	0x00300005	解码失败
47	HMEC_ERR_VIDEO_PARAM	0x00300006	解码参数 错误
48	HMEC_ERR_AUDIO_SPEEX_INIT	0x00300008	初始化音 频编解码

			器失败
49	HMEC_ERR_AUDIO_BUF_LEN	0x0030000A	传出缓冲区太小
50	HMEC_ERR_AUDIO_FIND_DECODER	0x0030000B	未知解码类型
51	HMEC_ERR_AUDIO_FIND_ENCODER	0x0030000C	未知编码类型
52	HMEC_ERR_AUDIO_CAP_INIT	0x0030000D	音频采集器初始化失败
53	HMEC_ERR_AUDIO_CAP_ALREADY_INIT	0x0030000E	音频采集器重复初始化
54	HMEC_ERR_AUDIO_CAPING	0x00300010	音频采集失败
55	HMEC_ERR_AUDIO_PLAYER_INIT	0x00300011	音频播放器初始化失败
56	HMEC_ERR_AUDIO_PLAYER_ALREADY_INIT	0x00300012	音频播放器重复初始化
57	HMEC_ERR_AUDIO_PLAYER_UNKNOWN_TYPE	0x00300013	未知音频类型
58	HMEC_ERR_AUDIO_PLAYER_SET_COOP_LEVEL	0x00300014	设置协作层失败
59	HMEC_ERR_AUDIO_PLAYER_CREATE_BUF	0x00300015	分配存储空间失败

60	HMEC_ERR_UTIL_INIT	0x00A00001	初始化失败
61	HMEC_ERR_UTIL_FILE_INVALID	0x00A00002	文件无效
62	HMEC_ERR_UTIL_FILE_NOT_OPEN	0x00A00003	无法打开文件
63	HMEC_ERR_UTIL_FILE_READ_FAIL	0x00A00004	读取文件失败
64	HMEC_ERR_UTIL_FILE_WRITE_FAIL	0x00A00005	写入文件失败
65	HMEC_ERR_UTIL_FILE_CREATE	0x00A00006	创建文件失败
66	HMEC_ERR_UTIL_RECORD_STOP	0x00A00007	停止录像
67	HMEC_ERR_UTIL_PARAM_INVALID	0x00A00008	参数无效
68	HMEC_ERR_UTIL_IP_NOT_FOUND	0x00A00009	IP 未能识别
69	HMEC_ERR_UTIL_SEND_MODE	0x00A0000A	发送模式错误
70	HMEC_ERR_UTIL_NOT_SUPPORT_IN_MODE	0x00A0000B	当前模式不支持

SDK 总体流程概览



流程简要说明：

- (1) 初始化 SDK，使用 SDK 的前提，对一些资源进行初始化，加载一些必要的库文件。
- (2) 登录设备或者先登录平台，从平台上获取必要的设备信息，然后连接设备。

登录平台后，首先需要获取整个设备列表，然后取得用户信息，根据用户信息的连接策略字段决定是否使用中转穿透，若需要使用，则需要获取所有设备的中转穿透信息。以上信息获取完毕之后方可取得一个设备，并建立连接。

- (3) 设备连接好之后便可对设备进行一系列的操作，目前预览所提供的功能是不断从设备获取视频或音频数据，在回调函数当中给出原始视音频数据，需由用户自行调用解码接口解码并显示。

- (4) 语音数据和视频数据类似，数据在回调中给出，需要用户自行解码并播放，编码音频数据也需要用户自行采集调用接口进行编码并发送。
- (5) 见 (4)。
- (6) 用户提供报警回调，产生报警时，设备会将报警信息推给 SDK，SDK 会在用户回调中将报警数据传出。
- (7) 云台控制只需要用户调用相应的控制接口，命令将传给前端设备。
- (8) 视音频编解码模块提供视频和音频处理编解码接口，其中视频只提供解码接口，音频提供编解码接口。视频解码接口的传出参数所有权都归 SDK 所有，用户不必关心存储空间释放问题，音频解码接口调用形式都是采用在外分配存储空间的形式，SDK 仅仅使用传进的存储空间，若空间不够，则返回失败并在传出参数中给出实际需要的空间大小，用户需要重新分配并再次调用接口函数。
- (9) 释放 SDK，释放资源。

接口描述

初始化相关

hm_sdk_init

初始化 SDK。

hm_result hm_sdk_init

()

Return Values

返回错误码。

可能返回的错误码:

- (1) HMEC_OK 成功。
- (2) HMEC_GEN_SDK_INIT 初始化失败。

Remarks

使用 SDK 之前必须调用此接口进行初始化。

hm_sdk_uninit

释放 SDK。

```
hm_result hm_sdk_uninit  
  
()
```

Remarks

使用完毕 SDK 后对其进行释放操作。

设备相关

hm_pu_login

连接设备。

```
hm_result hm_pu_login  
  
(  
  
    cpchar ip,  
  
    unsigned int port,  
  
    cpchar sn,  
  
    cpchar user,  
  
    cpchar pwd,  
  
    user_id* id
```


)

Parameters

ip	[in]	设备 IP 地址。
port	[in]	设备端口号。
sn	[in]	设备 SN 号。
user	[in]	用户名。
pwd	[in]	密码。
id	[in/out]	传出此设备句柄。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_GEN_CREATE_HANDLE	创建句柄失败。
(4) HMEC_ERR_PU_MAX_CONNECTION	最大连接数限制。
(5) HMEC_ERR_PU_BAD_USER	错误的用户名或密码。
(6) 系统网络错误码。	

Remarks

调用此函数成功时，id 会被赋值登录设备的句柄，之后在此设备上的操作均需使用此句柄。

See Also

[hm_pu_login_ex](#), [hm_pu_logout](#)

hm_pu_login_ex

连接设备扩展接口，使用节点句柄连接。

```
hm_result hm_pu_login_ex
(
    node_handle handle,
    user_id* id
)
```

Parameters

handle	[in]	设备句柄。
id	[in/out]	传出此设备句柄。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_GEN_CREATE_HANDLE	创建句柄失败。
(4) HMEC_ERR_PU_MAX_CONNECTION	最大连接数限制。
(5) HMEC_GEN_URL_INVALID	获取的资源无效。
(6) HMEC_ERR_PU_BAD_USER	错误的用户名。
(7) 系统网络错误码。	

Remarks

设备句柄的来源是设备列表，调用此接口前必须已经获取到了设备列表，直接将设备列表的 Device 或 DVS 属性节点取出作为其第一个参数，即可自动登录相应设备，登录时总是先尝试直连，若失败则根据配置策略进行下一步操作。若连接成功，id 会被赋值登录

设备的句柄，之后在此设备上的操作均需使用此句柄。

See Also

[hm_pu_login](#), [hm_pu_logout](#)

hm_pu_set_net_cb

设置底层网络错误回调。

```
hm_result hm_pu_set_net_cb
(
    user_id id,
    cb_pu_net cb,
    user_data data
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
cb	[in]	接收网络错误回调。

函数签名式：

```
void (user_data data, hm_result result, uint32 ct)
    data      [out]    同此接口的 data，原样传出。
    result    [out]    错误码。
    ct        [out]    当前连接类型。
```

data	[in]	用户数据。
------	------	-------

hm_pu_logout

断开设备。

```
hm_result hm_pu_logout
```

```
(
```

```
    user_id id
```

```
)
```

Parameters

id [in] hm_pu_login 或 hm_pu_login_ex 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |

Remarks

调用此接口后，hm_pu_login 或 hm_pu_login_ex 返回之句柄即失效，此时不应再用此句柄调用其他设备相关接口。

请确保此 id 已进行完所有相关操作再调用 hm_pu_logout 释放句柄。

See Also

[hm_pu_login](#), [hm_pu_login_ex](#)

hm_pu_get_device_info

获取设备信息。

```
hm_result hm_pu_get_device_info
(
    user_id id,
    P\_DEVICE\_INFO device
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
device	[in/out]	由用户创建的 DEVICE_INFO 结构，此接口负责填充。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_ERR_PU_NOT_LOGIN	未认证。
(4) HMEC_ERR_PU_CHANNEL_CLOSED	当前通道关闭。
(5) HMEC_ERR_PU_CHANNEL_ERR	错误的通道号。
(6) 其他网络错误码。	

See Also

[hm_pu_login](#), [hm_pu_login_ex](#)

hm_pu_open_video

创建实时视频句柄。

```
hm_result hm_pu_open_video
(
    user_id id,
    OPEN\_VIDEO\_PARAM param,
    video_handle* handle
)
```

Parameters

user	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
param	[in]	打开视频相关参数。
handle	[in/out]	返回视频播放句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|---------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_GEN_CREATE_HANDLE | 创建句柄失败。 |

Remarks

返回的视频播放句柄可用于其他操作，如 hm_start_video、hm_stop_video 等。

See Also

[hm_pu_login](#), [hm_pu_login_ex](#), [hm_pu_start_video](#), [hm_pu_stop_video](#),
[hm_pu_close_video](#)

hm_pu_start_video

打开实时视频。

```
hm_result hm_pu_start_video
(
    video_handle handle,
    P\_OPEN\_VIDEO\_RES res
)
```

Parameters

handle	[in]	hm_pu_open_video 之返回句柄。
res	[in/out]	请求视频返回参数，需用户自行提供存储空间。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_ERR_PU_BUSY	设备忙。
(4) HMEC_ERR_PU_BAD_PARAM	错误的参数。
(5) HMEC_ERR_PU_NOT_LOGIN	未认证。
(6) HMEC_ERR_PU_CHANNEL_USED	通道忙。
(7) HMEC_ERR_PU_CHANNEL_CLOSED	通道关闭。
(8) HMEC_ERR_PU_CHANNEL_ERR	错误的通道。
(9) HMEC_ERR_PU_OPERATION_ABORT	操作中断。
(10) 其他网络错误码。	

Remarks

调用此接口后，接收数据回调会不断被调用，请确保回调已被正确设置。

See Also

[hm_pu_open_video](#)

hm_pu_force_iframe

强制请求关键帧。

```
hm_result hm_pu_force_iframe
(
    video_handle handle,
)
```

Parameters

handle [in] hm_pu_open_video 之返回句柄。

Return Values

返回错误码。

Remarks

此接口用于向设备强制请求一帧关键帧。

hm_pu_stop_video

停止实时视频。


```
hm_result hm_pu_stop_video  
  
(  
  
    video_handle handle  
  
)
```

Parameters

handle [in] hm_pu_open_video 之返回句柄。

Return Value

返回错误码。

可能返回的错误码：

- (1) HMEC_OK 成功。
- (2) HMEC_GEN_PARAM_INVALID 参数错误。

Remarks

此接口需在 hm_close_video 之前被调用，以停止接收视频数据。

See Also

[hm_pu_start_video](#)

hm_pu_close_video

释放实时视频句柄。

```
hm_result hm_pu_close_video  
  
(  
  
    video_handle handle
```

)

Parameters

handle [in] hm_pu_open_video 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|---------------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |

Remarks

调用此接口后，视频句柄即失效，此时不应再用此句柄调用其他视频操作相关接口。

See Also

[hm_pu_open_video](#)

hm_pu_open_audio

创建实时音频句柄。

```
hm_result hm_pu_open_audio
(
    user_id user,
    P\_OPEN\_AUDIO\_PARAM param,
    P\_OPEN\_AUDIO\_RES res
```

```
audio_handle* handle
)
```

Parameters

user	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
param	[in]	请求音频输入参数结构体。
res	[in/out]	请求音频参数输出。
handle	[in/out]	返回音频句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|---------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_GEN_CREATE_HANDLE | 创建句柄失败。 |

Remarks

若调用成功，handle 会赋值音频句柄，对音频的后续操作都需使用此句柄。

See Also

[hm_pu_login](#), [hm_pu_login_ex](#), [hm_pu_start_audio](#), [hm_pu_stop_audio](#),
[hm_pu_close_audio](#)

hm_pu_start_audio

打开实时音频。

```
hm_result hm_pu_start_audio
```

```
(  
  
    audio_handle handle,  
  
)
```

Parameters

handle [in] hm_pu_open_audio 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|---------------------------------|--------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_BAD_PARAM | 错误的参数。 |
| (5) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (6) HMEC_ERR_PU_CHANNEL_USED | 通道忙。 |
| (7) HMEC_ERR_PU_CHANNEL_CLOSED | 通道关闭。 |
| (8) HMEC_ERR_PU_CHANNEL_ERR | 错误的通道。 |
| (9) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (10) 其他网络错误码。 | |

Remarks

调用此接口后，音频回调会不断收到数据，请确保在此之前已经设置好音频回调。

See Also

[hm_pu_open_audio](#)

hm_pu_stop_audio

停止实时音频。

```
hm_result hm_pu_stop_audio  
(  
    audio_handle handle  
)
```

Parameters

handle [in] hm_pu_open_audio 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|---------------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (4) 其他网络错误码 | |

Remarks

此接口需在 hm_close_audio 之前被调用，以停止接收音频数据。

See Also

[hm_pu_start_audio](#)

hm_pu_close_audio

关闭实时音频句柄。

```
hm_result hm_pu_close_audio
(
    audio_handle handle
)
```

Parameters

handle [in] hm_pu_open_audio 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- (1) HMEC_OK 成功。
- (2) HMEC_GEN_PARAM_INVALID 参数错误。

Remarks

调用此接口后，音频句柄即失效，此时不应再用此参数调用其他音频操作相关接口。

See Also

[hm_pu_open_audio](#)

hm_pu_open_talk

创建对讲句柄。

```
hm_result hm_pu_open_talk
(
```

```

    user_id id,

    P\_OPEN\_TALK\_PARAM param,

    talk_handle* handle

)

```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
param	[in]	请求对讲输入参数。
handle	[in/out]	返回对讲句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|---------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_GEN_CREATE_HANDLE | 创建句柄失败。 |

Remarks

若调用成功，handle 会被赋值对讲句柄，对讲的后续操作均需使用此句柄。

See Also

[hm_pu_login](#), [hm_pu_login_ex](#), [hm_pu_start_talk](#), [hm_pu_stop_talk](#),
[hm_pu_close_talk](#)

hm_pu_send_talk_data

发送对讲数据。

```
hm_result hm_pu_send_talk_data
(
    talk_handle handle,
    P_FRAME_DATA frame
)
```

Parameters

handle	[in]	hm_pu_open_talk 之返回句柄。
frame	[in]	需要发送的音频数据信息。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

hm_pu_start_talk

打开对讲。

```
hm_result hm_pu_start_talk
(
    talk_handle handle,
)
```


Parameters

handle [in] hm_pu_open_talk 之返回值。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|---------------------------------|---------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_BAD_PARAM | 错误的参数。 |
| (5) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (6) HMEC_ERR_PU_CHANNEL_USED | 通道忙。 |
| (7) HMEC_ERR_PU_CHANNEL_CLOSED | 通道关闭。 |
| (8) HMEC_ERR_PU_CHANNEL_ERR | 错误的通道。 |
| (9) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (10) HMEC_ERR_PU_UNSUPPORT | 不支持的参数。 |
| (11) 其他网络错误码。 | |

Remarks

调用此接口后，方可发送对讲数据。

See Also

[hm_pu_open_talk](#), [hm_pu_stop_talk](#)

hm_pu_stop_talk

停止对讲。

```
hm_result hm_pu_stop_talk
```

```
(
    talk_handle handle
)
```

Parameters

handle [in] hm_pu_open_talk 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|---------------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (4) 其他网络错误码。 | |

Remarks

此接口需在 hm_close_talk 之前调用，以停止发送语音数据。

See Also

[hm_pu_open_talk](#), [hm_pu_start_talk](#)

```
hm_pu_close_talk
```

关闭对讲句柄。

```
hm_result hm_pu_close_talk
```

```
(
    talk_handle handle
)
```

Parameters

handle [in] hm_pu_open_talk 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- (1) HMEC_OK 成功。
- (2) HMEC_GEN_PARAM_INVALID 参数错误。

Remarks

调用此接口后，对讲句柄即被释放，此时不应再使用此句柄做其他对讲操作。

See Also

[hm_pu_open_talk](#)

hm_pu_remote_capture_pic

远程抓图。

```
hm_result hm_pu_remote_capture_pic
(
    user_id id,
```

```
P_REMOTE_CAPTURE_PIC_PARAM param  
)
```

Parameters

- id [in] hm_pu_login 或 hm_pu_login_ex 之返回句柄。
- param [in] 远程抓图参数。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------------|--------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_DISK_NOT_EXIST | 存储不存在。 |
| (4) HMEC_ERR_PU_DISK_FULL | 存储满。 |
| (5) HMEC_ERR_PU_BUSY | 设备忙。 |
| (6) HMEC_ERR_PU_BAD_PARAM | 错误的参数。 |
| (7) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (8) HMEC_ERR_PU_CHANNEL_USED | 通道忙。 |
| (10) HMEC_ERR_PU_CHANNEL_CLOSED | 通道关闭。 |
| (11) HMEC_ERR_PU_CHANNEL_ERR | 错误的通道。 |
| (12) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (13) 其他网络错误码。 | |

Remarks

此接口用于在设备 SD 卡上保存一帧图像。

See Also

[hm_pu_login](#), [hm_pu_login_ex](#)

hm_pu_ptz_control

云台控制。

```
hm_result hm_pu_ptz_control
(
    user_id user,
    uint32 channel,
    PTZ\_COMMAND ptz_cmd,
    int32 speed
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
channel	[in]	设备通道号。
ptz_cmd	[in]	云台相关控制命令。
speed	[in]	相关操作的速度，其范围为 0-10。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_ERR_PU_BUSY	设备忙。
(4) HMEC_ERR_PU_BAD_PARAM	错误的参数。

- (5) HMEC_ERR_PU_NOT_LOGIN 未认证。
- (6) HMEC_ERR_PU_CHANNEL_USED 通道忙。
- (7) HMEC_ERR_PU_CHANNEL_CLOSED 通道关闭。
- (8) HMEC_ERR_PU_CHANNEL_ERR 错误的通道。
- (9) HMEC_ERR_PU_OPERATION_ABORT 操作中断。
- (10) 其他网络错误码。

See Also

[hm_pu_login](#), [hm_pu_login_ex](#)

hm_pu_ptz_set_preset

设置预置位。

```
hm_result hm_pu_ptz_set_preset
(
    user_id id,
    uint32 channel,
    uint8 index,
    cpchar preset_name
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
channel	[in]	通道号。
index	[in]	预置位编号。
preset_name	[in]	预置位名称。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_ERR_PU_BUSY	设备忙。
(4) HMEC_ERR_PU_BAD_PARAM	错误的参数。
(5) HMEC_ERR_PU_NOT_LOGIN	未认证。
(6) HMEC_ERR_PU_CHANNEL_USED	通道忙。
(7) HMEC_ERR_PU_CHANNEL_CLOSED	通道关闭。
(8) HMEC_ERR_PU_CHANNEL_ERR	错误的通道。
(9) HMEC_ERR_PU_OPERATION_ABORT	操作中断。
(10) 其他网络错误码。	

See Also

[hm_pu_login](#), [hm_pu_login_ex](#)

hm_pu_ptz_clr_preset

控制预置位。

```
hm_result hm_pu_ptz_clr_preset
(
    user_id id,
    uint32 channel,
    uint8 index
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
channel	[in]	通道号。
index	[in]	预置位编号。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_ERR_PU_BUSY	设备忙。
(4) HMEC_ERR_PU_BAD_PARAM	错误的参数。
(5) HMEC_ERR_PU_NOT_LOGIN	未认证。
(6) HMEC_ERR_PU_CHANNEL_USED	通道忙。
(7) HMEC_ERR_PU_CHANNEL_CLOSED	通道关闭。
(8) HMEC_ERR_PU_CHANNEL_ERR	错误的通道。
(9) HMEC_ERR_PU_OPERATION_ABORT	操作中断。
(10) 其他网络错误码。	

See Also

[hm_pu_login](#), [hm_pu_login_ex](#)

hm_pu_ptz_goto_preset

移动到预置位。

hm_result hm_pu_ptz_goto_preset


```
(  
  
    user_id id,  
  
    uint32 channel,  
  
    uint8 index  
  
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
channel	[in]	通道号。
index	[in]	预置位编号。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_ERR_PU_BUSY	设备忙。
(4) HMEC_ERR_PU_BAD_PARAM	错误的参数。
(5) HMEC_ERR_PU_NOT_LOGIN	未认证。
(6) HMEC_ERR_PU_CHANNEL_USED	通道忙。
(7) HMEC_ERR_PU_CHANNEL_CLOSED	通道关闭。
(8) HMEC_ERR_PU_CHANNEL_ERR	错误的通道。
(9) HMEC_ERR_PU_OPERATION_ABORT	操作中断。
(10) 其他网络错误码。	

See Also

[hm_pu_login](#), [hm_pu_login_ex](#)

hm_pu_ptz_cruise

预置位巡航。

```
hm_result hm_pu_ptz_cruise
(
    user_id id,
    PTZ_INTERVAL type,
    uint32 channel,
    uint32 speed
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
type	[in]	云台命令类型。
channel	[in]	通道号。
speed	[in]	速度值。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_ERR_PU_BUSY	设备忙。
(4) HMEC_ERR_PU_BAD_PARAM	错误的参数。

- | | |
|---------------------------------|--------|
| (5) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (6) HMEC_ERR_PU_CHANNEL_USED | 通道忙。 |
| (7) HMEC_ERR_PU_CHANNEL_CLOSED | 通道关闭。 |
| (8) HMEC_ERR_PU_CHANNEL_ERR | 错误的通道。 |
| (9) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (10) 其他网络错误码。 | |

See Also

[hm_pu_login](#), [hm_pu_login_ex](#)

hm_pu_ptz_focus_up

正向调整聚焦。

```
hm_result hm_pu_ptz_focus_up
(
    user_id user,
    uint32 channel,
    int32 speed
)
```

Parameters

- | | | |
|---------|------|-------------------------------------|
| user | [in] | hm_pu_login 或 hm_pu_login_ex 之返回句柄。 |
| channel | [in] | 设备通道号。 |
| speed | [in] | 相关操作的速度，其范围为 0-10。 |

Return Values

- | | |
|---------------------------------|--------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_BAD_PARAM | 错误的参数。 |
| (5) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (6) HMEC_ERR_PU_CHANNEL_USED | 通道忙。 |
| (7) HMEC_ERR_PU_CHANNEL_CLOSED | 通道关闭。 |
| (8) HMEC_ERR_PU_CHANNEL_ERR | 错误的通道。 |
| (9) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (10) 其他网络错误码。 | |

Remarks

调用此接口调整前端设备的聚焦水平。

See Also

[hm_pu_login](#), [hm_pu_login_ex](#), [hm_pu_ptz_focus_down](#)

hm_pu_ptz_focus_down

反向调整聚焦。

```
hm_result hm_pu_ptz_focus_down
(
    user_id user,
    uint32 channel,
    int32 speed
)
```

Parameters

user	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
channel	[in]	设备通道号。
speed	[in]	相关操作的速度，其范围为 0-10。

Return Values

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_ERR_PU_BUSY	设备忙。
(4) HMEC_ERR_PU_BAD_PARAM	错误的参数。
(5) HMEC_ERR_PU_NOT_LOGIN	未认证。
(6) HMEC_ERR_PU_CHANNEL_USED	通道忙。
(7) HMEC_ERR_PU_CHANNEL_CLOSED	通道关闭。
(8) HMEC_ERR_PU_CHANNEL_ERR	错误的通道。
(9) HMEC_ERR_PU_OPERATION_ABORT	操作中断。
(10) 其他网络错误码。	

Remarks

调用此接口调整前端设备的聚焦水平。

See Also

[hm_pu_login](#), [hm_pu_login_ex](#), [hm_pu_ptz_focus_up](#)

hm_pu_open_alarm

创建报警句柄。

```
hm_result hm_pu_open_alarm
(
    user_id id,
    P\_OPEN\_ALARM\_PARAM param,
    alarm_handle* handle
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
param	[in]	报警参数。
handle	[in/out]	传出报警句柄。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_GEN_CREATE_HANDLE	创建句柄失败。

Remarks

调用此接口后，可以创建报警主动上传对象，此后的相关操作均需使用此对象。

See Also

[hm_pu_login](#), [hm_pu_login_ex](#)

hm_pu_start_alarm

开启报警。

```
hm_result hm_pu_start_alarm  
(  
    alarm_handle handle  
)
```

Parameters

handle [in] hm_pu_open_alarm 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

hm_pu_stop_alarm

关闭报警。

```
hm_result hm_pu_stop_alarm  
(  
    alarm_handle handle  
)
```

Parameters

handle [in] hm_pu_open_alarm 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

hm_pu_close_alarm

释放报警句柄。

```
hm_result hm_pu_close_alarm
(
    alarm_handle handle
)
```

Parameters

handle [in] hm_pu_open_alarm 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口后，报警对象即被释放，此时不应再用此对象进行其他报警相关操作。

See Also

[hm_pu_open_alarm](#)

hm_pu_open_record

创建远程录像句柄。

```
hm_result hm_pu_open_record
(
    user_id id,
    int32 chn,
    record_handle* handle
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
chn	[in]	通道号。
handle	[in/out]	返回录像句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|---------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_GEN_CREATE_HANDLE | 创建句柄失败。 |

Remarks

若此接口调用成功，则 handle 会赋值录像句柄，此后的录像操作均需使用此句柄。

See Also

[hm_pu_login](#), [hm_pu_login_ex](#)

hm_pu_start_record

开启远程录像。

```
hm_result hm_pu_start_record
(
    record_handle handle,
    uint32 channel
)
```

Parameters

- handle[in]hm_pu_open_record 之返回句柄。
- channel[in]通道号。

Return Values

返回错误码。

可能返回的错误码：

- (1) HMEC_OK成功。
- (2) HMEC_GEN_PARAM_INVALID参数错误。
- (3) HMEC_ERR_PU_DISK_NOT_EXIST存储不存在。

- | | |
|----------------------------------|--------|
| (4) HMEC_ERR_PU_DISK_FULL | 存储满。 |
| (5) HMEC_ERR_PU_BUSY | 设备忙。 |
| (6) HMEC_ERR_PU_BAD_PARAM | 错误的参数。 |
| (7) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (8) HMEC_ERR_PU_CHANNEL_USED | 通道忙。 |
| (10) HMEC_ERR_PU_CHANNEL_CLOSED | 通道关闭。 |
| (11) HMEC_ERR_PU_CHANNEL_ERR | 错误的通道。 |
| (12) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (13) 其他网络错误码。 | |

Remarks

此接口在设备端开启录像，设备会将视频文件保存在设备的 SD 卡上。

See Also

[hm_pu_open_record](#)

hm_pu_stop_record

关闭远程录像。

```
hm_result hm_pu_stop_record
(
    record_handle handle,
    uint32 channel
)
```

Parameters

handle	[in]	hm_pu_open_record 之返回句柄。
channel	[in]	通道号。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|---------------------------------|--------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_BAD_PARAM | 错误的参数。 |
| (5) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (6) HMEC_ERR_PU_CHANNEL_ERR | 错误的通道。 |
| (7) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (8) 其他网络错误码 | |

Remarks

调用此接口通知设备端停止录像。

See Also

[hm_pu_start_record](#)

hm_pu_close_record

释放远程录像句柄。

hm_result hm_pu_close_record

(

record_handle handle

```
)
```

Parameters

handle [in] hm_pu_open_record 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- (1) HMEC_OK 成功。
- (2) HMEC_GEN_PARAM_INVALID 参数错误。

Remarks

调用此接口后，录像句柄即失效，此时不应再用此接口进行其他录像操作。

See Also

[hm_pu_open_record](#), [hm_pu_start_record](#), [hm_pu_stop_record](#)

hm_pu_find_file

创建录像文件查找句柄。

```
hm_result hm_pu_find_file
(
    user_id id,
    P\_FIND\_FILE\_PARAM param,
    find_file_handle* handle
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
param	[in]	FIND_FILE_PARAM 结构，由用户填充，接口通过该变量中的信息查询录像文件。
handle	[in/out]	返回文件查找句柄。

Return Values

返回错误码。

可能返回的错误码：

- (1) HMEC_OK

成功。
- (2) HMEC_GEN_PARAM_INVALID

参数错误。

Remarks

利用此接口返回的文件查找句柄可在设备端查询所有的录像文件，回放时这些信息可能会有用处。

See Also

[hm_pu_login](#), [hm_pu_login_ex](#)

hm_pu_find_next_file

查找下一个录像文件。

```
hm_result hm_pu_find_next_file
(
    find_file_handle handle,
```

```
P_FIND_FILE_DATA data
)
```

Parameters

handle	[in]	hm_pu_find_file 之返回句柄。
data	[out]	返回搜索到的文件信息。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|-----------------------------|---------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_GEN_FIND_FILE_OVER | 已查找到末尾。 |

Remarks

反复调用该接口已查到下一个文件，当返回 HMEC_GEN_FIND_FILE_OVER 时标明文件查到已到末尾，此时应结束查找。

See Also

[hm_pu_find_file](#)

hm_pu_close_find_file

释放录像文件查找句柄。

```
hm_result hm_pu_close_find_file
(
```

```
find_file_handle handle
)
```

Parameters

handle [in] hm_pu_find_file 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- (1) HMEC_OK 成功。
- (2) HMEC_GEN_PARAM_INVALID 参数错误。

Remarks

调用此接口后，文件查找对象即被释放，此时不应再使用此对象调用其他文件查找接口。

hm_pu_delete_record_file

删除录像文件。

```
hm_result hm_pu_delete_record_file
(
    user_id id,
    P\_DELETE\_RECORD\_FILE\_PARAM param
)
```


Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
param	[in]	删除文件相关信息。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_ERR_PU_BUSY	设备忙。
(4) HMEC_ERR_PU_BAD_PARAM	错误的参数。
(5) HMEC_ERR_PU_NOT_LOGIN	未认证。
(6) HMEC_ERR_PU_CHANNEL_USED	通道忙。
(7) HMEC_ERR_PU_CHANNEL_ERR	错误的通道。
(8) HMEC_ERR_PU_OPERATION_ABORT	操作中断。
(9) 其他网络错误码。	

See Also

[hm_pu_login](#), [hm_pu_login_ex](#)

hm_pu_open_playback

创建远程回放句柄。

```
hm_result hm_pu_open_playback
(
    user_id id,
    P\_PLAYBACK\_PARAM param,
```

[P_PLAYBACK_RES](#) res

play_back_handle* handle

)

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
param	[in]	播放参数，接口根据此参数回调数据。
res	[out]	返回视频相关参数。
handle	[in/out]	返回录像回放句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|---------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_GEN_CREATE_HANDLE | 创建句柄失败。 |

Remarks

若此接口执行成功，handle 会被赋值回放句柄，此后的回放相关操作均需使用此句柄。

See Also

[hm_pu_login](#), [hm_pu_login_ex](#)

hm_pu_start_playback

开启远程回放。

```
hm_result hm_pu_start_playback
(
    playback_handle handle,
)
```

Parameters

handle [in] hm_pu_open_playback 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|---------------------------------|--------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_BAD_PARAM | 错误的参数。 |
| (5) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (6) HMEC_ERR_PU_CHANNEL_USED | 通道忙。 |
| (7) HMEC_ERR_PU_CHANNEL_CLOSED | 通道关闭。 |
| (8) HMEC_ERR_PU_CHANNEL_ERR | 错误的通道。 |
| (9) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (10) 其他网络错误码。 | |

Remarks

调用此函数回放回调即会被调用，请确保在此之前已设置好回放回调。

See Also

[hm_pu_open_playback](#)

hm_pu_stop_playback

关闭远程回放。

hm_result hm_pu_stop_playback

(

playback_handle handle

)

Parameters

 handle [in] hm_pu_open_playback 之返回句柄。

Return Values

 返回错误码。

 可能返回的错误码：

- | | |
|---------------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (5) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (6) 其他网络错误码。 | |

Remarks

 此接口停止回放数据回调。

See Also

[hm_pu_start_playback](#)

hm_pu_pause_playback

暂停远程回放。

```
hm_result hm_pu_pause_playback
(
    playback_handle handle
)
```

Parameters

handle [in] hm_pu_open_playback 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|---------------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (5) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (6) 其他网络错误码。 | |

Remarks

此接口暂停回放数据回调。

See Also

[hm_pu_start_playback](#), [hm_pu_stop_playback](#)

hm_pu_resume_playback

恢复远程回放。

```
hm_result hm_pu_resume_playback
(
    playback_handle handle
)
```

Parameters

handle [in] hm_pu_open_playback 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|---------------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (5) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (6) 其他网络错误码。 | |

Remarks

此接口恢复回放数据回调。

See Also

[hm_pu_pause_playback](#)

hm_pu_step_playback

单帧回放。

```
hm_result hm_pu_step_playback
(
    playback_handle handle
)
```

Parameters

handle [in] hm_pu_open_playback 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|---------------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (5) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (6) 其他网络错误码。 | |

See Also

[hm_pu_start_playback](#)

hm_pu_close_playback

释放远程回放句柄。

```
hm_result hm_pu_close_playback
(
    playback_handle handle
)
```

Parameters

handle [in] hm_pu_open_playback 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|---------------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (5) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (6) 其他网络错误码。 | |

Remarks

调用此接口，回放句柄即被释放，此时不应再使用此句柄进行其他回放相关操作。

See Also

[hm_pu_open_playback](#)

hm_pu_open_get_file

创建文件下载句柄。

```
hm_result hm_pu_open_get_file
(
    user_id id,
    P\_GET\_FILE\_PARAM param,
    P\_GET\_FILE\_RES res,
    get_file_handle* handle
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
param	[in]	获取文件信息，由用户填充。
res	[out]	返回下载文件的信息，用户得到该值可根据文件大小进行文件保存等相关操作。
handle	[in/out]	返回文件下载句柄。

Return Values

返回错误码。
可能返回的错误码：

- | | |
|----------------------------|---------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_GEN_CREATE_HANDLE | 创建句柄失败。 |

Remarks

若调用成功，则 handle 会被赋值文件下载句柄，后续文件下载操作均需使用此句柄。

See Also

[hm_pu_login](#), [hm_pu_login_ex](#)

hm_pu_start_get_file

开启文件下载。

```
hm_result hm_pu_start_get_file
(
    get_file_handle handle,
)
```

Parameters

handle [in] hm_pu_open_get_file 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|-------------|-----|
| (1) HMEC_OK | 成功。 |
|-------------|-----|

- | | |
|---------------------------------|--------|
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_BAD_PARAM | 错误的参数。 |
| (5) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (6) HMEC_ERR_PU_CHANNEL_USED | 通道忙。 |
| (7) HMEC_ERR_PU_CHANNEL_CLOSED | 通道关闭。 |
| (8) HMEC_ERR_PU_CHANNEL_ERR | 错误的通道。 |
| (9) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (10) 其他网络错误码。 | |

Remarks

param 参数提供欲下载文件所在通道、文件名和偏移量，若设备支持断点续传，则可以设置相应偏移（取数据地点距离文件头的字节单位长度），从偏移处开始取得视频文件数据。接口调用成功时，res 被赋值相应结构指针，指明文件的总大小（以字节为单位），下载时可使用该值确定文件是否已下载完毕。

See Also

[hm_pu_open_get_file](#)

hm_pu_cancel_get_file

关闭文件下载。

```
hm_result hm_pu_cancel_get_file
(
    get_file_handle handle
)
```

Parameters

handle [in] hm_pu_open_get_file 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|---------------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (5) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (6) 其他网络错误码。 | |

Remarks

调用此接口取消文件下载回调，若非正常结束回调（文件下载完毕），强烈建议先调用此接口取消下载，再调用 hm_close_get_file。

See Also

[hm_pu_open_get_file](#), [hm_pu_start_get_file](#), [hm_pu_close_get_file](#)

hm_pu_close_get_file

关闭文件下载句柄。

hm_result hm_pu_close_get_file

(

get_file_handle handle

handle [in] hm_pu_open_get_file 之返回句柄。

返回错误码。

可能返回的错误码:

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

调用此接口后，下载文件句柄即被释放，此时不应再用此句柄进行其他下载相关操作。

hm_pu_open_get_file

创建图片文件查找句柄。

```
hm_result hm_pu_find_picture
(
    user_id id,
    P_FIND_PICTURE_PARAM param,
    find_picture_handle* handle
```

)

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
param	[in]	查找图片参数，由用户填充。
handle	[in/out]	返回图片查找句柄。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_ERR_PU_BUSY	设备忙。
(4) HMEC_ERR_PU_BAD_PARAM	错误的参数。
(5) HMEC_ERR_PU_NOT_LOGIN	未认证。
(6) HMEC_ERR_PU_CHANNEL_USED	通道忙。
(7) HMEC_ERR_PU_CHANNEL_CLOSED	通道关闭。
(8) HMEC_ERR_PU_CHANNEL_ERR	错误的通道。
(9) HMEC_ERR_PU_OPERATION_ABORT	操作中断。
(10) 其他网络错误码。	

See Also

[hm_pu_login](#), [hm_pu_login_ex](#), [hm_pu_find_next_picture](#)

hm_pu_find_next_picture

查找下一个图片文件。

hm_result hm_pu_find_next_picture

```
(
    find_picture_handle handle,
    P\_FIND\_PICTURE\_DATA data
)
```

Parameters

handle	[in]	hm_pu_find_picture 之返回句柄。
data	[out]	查找到的图片信息。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_GEN_FIND_FILE_OVER	已查找到文件末尾。

Remarks

反复调用该接口已获取所有保存的图片信息，直到接口返回 HMEC_GEN_FIND_FILE_OVER 为止。

See Also

[hm_pu_find_picture](#)

hm_pu_close_find_picture

释放图片文件查找句柄。

```
hm_result hm_pu_close_find_picture
(
    find_picture_handle handle
)
```

Parameters

handle [in] hm_pu_find_picture 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口后，图片查找句柄即被释放，此时不应再用此句柄进行其他图片查找相关操作。

See Also

[hm_pu_find_picture](#)

hm_pu_delete_picture

删除设备图片。

```
hm_result hm_pu_delete_picture
```



```
(  
  
    user_id id,  
  
    P\_DELETE\_PICTURE\_PARAM param  
  
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
param	[in]	欲删除图片的相关信息，由用户填充。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_ERR_PU_BUSY	设备忙。
(4) HMEC_ERR_PU_BAD_PARAM	错误的参数。
(5) HMEC_ERR_PU_NOT_LOGIN	未认证。
(6) HMEC_ERR_PU_CHANNEL_USED	通道忙。
(7) HMEC_ERR_PU_CHANNEL_CLOSED	通道关闭。
(8) HMEC_ERR_PU_CHANNEL_ERR	错误的通道。
(9) HMEC_ERR_PU_OPERATION_ABORT	操作中断。
(10) 其他网络错误码。	

See Also

[hm_pu_login](#), [hm_pu_login_ex](#)

hm_pu_open_get_picture

创建远程图片下载句柄。

```
hm_result hm_pu_open_get_picture
(
    user_id id,
    P\_GET\_PICTURE\_PARAM param,
    P\_GET\_PICTURE\_RES res
    get_picture_handle* handle
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
param	[in]	下载图片的相关信息，由用户填充。
res	[out]	返回图片的相关信息。
handle	[in/out]	返回图片下载句柄。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_GEN_CREATE_HANDLE	创建句柄失败。

Remarks

若函数调用成功，handle 会被赋值下载图片句柄，下载图片的后续操作均需用到此句柄。

See Also

[hm_pu_login](#), [hm_pu_login_ex](#)

hm_pu_start_get_picture

开启远程图片下载。

```
hm_result hm_pu_start_get_picture  
(  
    get_picture_handle handle,  
)
```

Parameters

handle [in] hm_pu_open_get_picture 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

用户需填充 param 结构，最主要的参数是欲获取的图片名称。获取到的图片相关信息通过 res 参数返回，包括图片的大小等。此接口一旦调用，接收图片数据回调便开始调用，请在调用此接口之前设置好数据回调。

See Also

[hm_pu_open_get_picture](#)

hm_pu_cancel_get_picture

关闭远程图片下载。

```
hm_result hm_pu_cancel_get_picture
```

```
(
```

```
    get_picture_handle handle
```

```
)
```

Parameters

handle [in] hm_pu_open_get_picture 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|---------------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (5) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (6) 其他网络错误码。 | |

Remarks

调用此接口取消图片下载回调，若非正常结束回调（图片下载完毕），强烈建议先

调用此接口取消下载，再调用 `hm_close_get_picture`。

See Also

[hm_pu_close_get_picture](#)

hm_pu_close_get_picture

释放远程图片下载句柄。

```
hm_result hm_pu_close_get_picture  
(  
    get_picture_handle handle  
)
```

Parameters

`handle` [in] `hm_pu_open_get_picture` 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口后，图片下载句柄即失效，此时不应再使用此句柄进行图片下载相关操作。

See Also

[hm_pu_cancel_get_picture](#)

hm_pu_reboot

重新启动设备。

```
hm_result hm_pu_reboot
(
    user_id id
)
```

Parameter

id [in] hm_pu_login 或 hm_pu_login_ex 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|---------------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (5) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (6) 其他网络错误码。 | |

Remarks

调用此接口重启前端设备。

hm_pu_open_search_wifi

创建 WiFi 搜索句柄。

```
hm_result hm_pu_open_search_wifi
(
    user_id id,
    P_QUERY_WIFI_PARAM param,
    search_wifi_handle* handle
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
param	[in]	WiFi 搜索参数。
handle	[in/out]	返回 wifi 搜索句柄。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_GEN_CREATE_HANDLE	创建句柄失败。

Remarks

若接口调用成功，handle 会被赋值搜索 wifi 句柄，此后搜索 wifi 的相关操作均需使用此句柄。

See Also

[hm_pu_login](#), [hm_pu_login_ex](#)

hm_pu_set_wifi_callback

设置 WiFi 信息回调函数。

```
hm_result hm_pu_set_wifi_callback
(
    search_wifi_handle handle,
    cb_pu_wifi cb,
    user_data data
)
```

Parameters

- handle [in] hm_pu_open_search_wifi 之返回句柄。
- cb [in] 接收 wifi 信息回调。

函数签名式:

void (user_data data, WIFI_INFO wifi_info)

- data [out] 同此接口的 data，原样传出。
- wifi_info [out] wifi 信息。

data [in] 用户数据。

Return Values

返回错误码。

可能返回的错误码:

- (1) HMEC_OK 成功。
- (2) HMEC_GEN_PARAM_INVALID 参数错误。

See Also

[hm_pu_open_search_wifi](#), [hm_pu_start_search_wifi](#), [hm_pu_close_search_wifi](#)

hm_pu_start_search_wifi

开启 WiFi 搜索。

```
hm_result hm_pu_start_search_wifi  
  
(  
  
    search_wifi_handle  
  
)
```

Parameters

handle [in] hm_pu_open_search_wifi 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- (1) HMEC_OK 成功。
- (2) HMEC_GEN_PARAM_INVALID 参数错误。

Remarks

调用此接口后，接收 wifi 数据回调会开始调用，请确保调用此接口前 wifi 数据回调已设置妥当。

See Also

[hm_pu_open_search_wifi](#)

hm_pu_close_search_wifi

释放 WiFi 搜索句柄。

```
hm_result hm_pu_close_search_wifi  
(  
    search_wifi_handle  
)
```

Parameters

handle [in] hm_pu_open_search_wifi 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口后，wifi 搜索句柄即被释放，此时不应再用此句柄进行其他 wifi 搜索相关操作。

See Also

[hm_pu_open_search_wifi](#)

hm_pu_detect_upnp

检查 UPNP。

```
hm_result hm_pu_detect_upnp
(
    user_id id,
    P\_DETECT\_UPNP upnp
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
upnp	[out]	返回 UPNP 相关信息。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|---------------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (5) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (6) 其他网络错误码。 | |

hm_pu_format_sd

格式化设备 SD 卡。

```
hm_result hm_pu_format_sd
(
    user_id id,
    uint8* format_result
)
```

Parameters

- id [in] hm_pu_login 或 hm_pu_login_ex 之返回句柄。
- format_result [in/out] 格式化结果。

Return Values

返回错误码。

可能返回的错误码：

- (1) HMEC_OK 成功。
- (2) HMEC_GEN_PARAM_INVALID 参数错误。
- (3) HMEC_ERR_PU_DISK_NOT_EXIST 存储不存在。
- (4) HMEC_ERR_PU_BUSY 设备忙。
- (5) HMEC_ERR_PU_BAD_PARAM 错误的参数。
- (6) HMEC_ERR_PU_NOT_LOGIN 未认证。
- (7) HMEC_ERR_PU_OPERATION_ABORT 操作中断。
- (8) 其他网络错误码。

Remarks

调用此接口格式化前端设备 SD 卡，格式化结果由参数 `format_result` 返回。

hm_pu_restore_default_config

恢复设备默认配置。

```
hm_result hm_pu_restore_default_config
(
    user_id id,
    uint8* restore_result
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
restore_result	[in/out]	恢复默认配置结果

Return Values

返回错误码。

可能返回的错误码：

- | | |
|---------------------------------|--------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_BAD_PARAM | 错误的参数。 |
| (5) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (6) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (7) 其他网络错误码。 | |

Remarks

调用此接口使设备恢复到出厂设置。

hm_pu_get_system_info

获取设备系统信息。

```
hm_result hm_pu_get_system_info
(
    user_id id,
    P\_PT\_SYSTEM\_INFO system_info
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
system_info	[in]	返回设备系统信息。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_ERR_PU_BUSY	设备忙。
(4) HMEC_ERR_PU_BAD_PARAM	错误的参数。
(5) HMEC_ERR_PU_NOT_LOGIN	未认证。
(6) HMEC_ERR_PU_OPERATION_ABORT	操作中断。
(7) 其他网络错误码。	

Remarks

若接口调用成功，前端设备的系统信息通过参数 system_info 返回。

hm_pu_time_sync

设备与本地时间同步。

```
hm_result hm_pu_time_sync
(
    user_id id,
    uint32 time
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
time	[in]	同步时间。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|---------------------------------|--------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_BAD_PARAM | 错误的参数。 |
| (5) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (6) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |

(7) 其他网络错误码。

Remarks

此接口用于和设备端同步时间,调用此接口后设备端的时间和第二个参数指定的时间保持一致。

hm_pu_open_upgrade

创建设备升级句柄。

```
hm_result hm_pu_open_upgrade
(
    user_id id,
    P_HARD_UPDATE_PARAM param,
    upgrade_handle* handle
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
param	[in]	升级参数。
handle	[in/out]	传出升级句柄。

Return Values

返回错误码。

可能返回的错误码:

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。

(3) HMEC_GEN_CREATE_HANDLE 创建句柄失败。

Remarks

成功调用此函数，handle 会被赋值升级句柄，此后的升级操作均需使用此句柄。

See Also

[hm_pu_login](#), [hm_pu_login_ex](#)

hm_pu_start_upgrade

开启设备升级。

```
hm_result hm_pu_start_upgrade
(
    upgrade_handle handle,
)
```

Parameters

handle [in] hm_pu_open_upgrade 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|--------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_BAD_PARAM | 错误的参数。 |

- (5) HMEC_ERR_PU_NOT_LOGIN 未认证。
- (6) HMEC_ERR_PU_OPERATION_ABORT 操作中断。
- (7) 其他网络错误码

Remarks

用户需持有升级包，调用升级系列接口才有意义。

See Also

[hm_pu_open_upgrade](#)

hm_pu_send_upgrade_data

发送设备升级数据。

```
hm_result hm_pu_send_upgrade_data
(
    upgrade_handle handle,
    uint32 len,
    cpointer buf
)
```

Parameters

- | | | |
|--------|------|---------------------------|
| handle | [in] | hm_pu_open_upgrade 之返回句柄。 |
| len | [in] | 数据长度。 |
| buf | [in] | 数据缓冲区。 |

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

See Also

[hm_pu_open_upgrade](#), [hm_pu_start_upgrade](#)

hm_pu_cancel_upgrade

关闭设备升级。

```
hm_result hm_pu_cancel_upgrade
(
    upgrade_handle handle
)
```

Parameters

handle [in] hm_pu_open_upgrade 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|--------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_BAD_PARAM | 错误的参数。 |
| (5) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |

- (6) HMEC_ERR_PU_OPERATION_ABORT 操作中断。
- (7) 其他网络错误码。

See Also

[hm_pu_start_upgrade](#)

hm_pu_set_wifi_config

设置 WiFi 配置信息。

```
hm_result hm_pu_set_wifi_config
(
    user_id id,
    P\_CONFIG\_WIFI config
)
```

Parameters

- | | | |
|--------|------|-------------------------------------|
| id | [in] | hm_pu_login 或 hm_pu_login_ex 之返回句柄。 |
| config | [in] | WIFI 配置信息。 |

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|--------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_PU_BUSY | 设备忙。 |
| (4) HMEC_ERR_PU_BAD_PARAM | 错误的参数。 |

- (5) HMEC_ERR_PU_NOT_LOGIN 未认证。
- (6) HMEC_ERR_PU_OPERATION_ABORT 操作中断。
- (7) 其他网络错误码。

See Also

[hm_pu_get_wifi_config](#)

hm_pu_get_wifi_config

获取 WiFi 配置信息。

```
hm_result hm_pu_get_wifi_config
(
    user_id id,
    P\_CONFIG\_WIFI config
)
```

Parameters

- | | | |
|--------|-------|-------------------------------------|
| id | [in] | hm_pu_login 或 hm_pu_login_ex 之返回句柄。 |
| config | [out] | WIFI 配置信息。 |

Return Values

返回错误码。

可能返回的错误码：

- (1) HMEC_OK 成功。
- (2) HMEC_GEN_PARAM_INVALID 参数错误。
- (3) HMEC_ERR_PU_BUSY 设备忙。

- | | |
|---------------------------------|--------|
| (4) HMEC_ERR_PU_BAD_PARAM | 错误的参数。 |
| (5) HMEC_ERR_PU_NOT_LOGIN | 未认证。 |
| (6) HMEC_ERR_PU_OPERATION_ABORT | 操作中断。 |
| (7) 其他网络错误码。 | |

See Also

[hm_pu_set_wifi_config](#)

平台相关

hm_server_connect

登录平台。

```
hm_result hm_server_connect
(
    P_LOGIN_SERVER_INFO lsi,
    server_id* hserver,
    pchar err,
    int32 err_len
)
```

Parameters

- | | | |
|---------|-------|----------------------|
| lsi | [in] | 登录平台信息。 |
| hserver | [out] | 返回之句柄。 |
| err | [out] | 返回错误描述文本信息（UTF8 编码）。 |
| err_len | [in] | err 缓冲区可用长度。 |

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_CREATE_HANDLE	创建句柄失败。
(3) HMEC_ERR_SERVER_NETWORK	网络错误。
(4) HMEC_ERR_SERVER_COMMAND	命令错误。
(5) HMEC_ERR_SERVER_USER_OR_PWD	用户名或密码错误。
(6) HMEC_ERR_SERVER_ALREADY_LOGIN	重复登录。

Remarks

返回之句柄用于所有对平台的后续操作。

See Also

[hm_server_disconnect](#)

hm_server_disconnect

登出平台。

```
hm_result hm_server_disconnect
(
    server_id server
)
```

Parameters

server [in] hm_server_connect 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- (1) HMEC_OK 成功。
- (2) HMEC_GEN_PARAM_INVALID 参数错误。

Remarks

调用此接口后，平台句柄即失效，此时不应再用此参数调用其他平台操作相关接口。
如果需要对设备树进行操作，则须确保 hm_server_get_tree 先于此接口被调用，否则设备树会被释放。

See Also

[hm_server_connect](#), [hm_server_get_tree](#)

hm_server_get_device_list

获取设备列表。

```
hm_result hm_server_get_device_list
(
    server_id server
)
```

Parameters

server [in] hm_server_connect 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|-----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_SERVER_NETWORK | 网络错误。 |
| (4) HMEC_ERR_SERVER_COMMAND | 命令错误。 |

Remarks

调用此接口从平台获取分组和设备列表,内部将维护一棵树,用于以后的列表显示,取得设备并连接等环节。

See Also

[hm_server_connect](#)

hm_server_get_user_info

获取当前登录用户信息。

```
hm_result hm_server_get_user_info
(
    server_id server,
    P\_USER\_INFO uinfo
)
```

Parameters

server [in] hm_server_connect 之返回句柄。

uinfo [out] 用户信息结构。

Return Values

成功返回 0，否则返回非 0 值。

See Also

[hm_server_connect](#)

hm_server_get_transfer_info

获取中转穿透信息。

```
hm_result hm_server_get_transfer_info
(
    server_id server
)
```

Parameters

server [in] hm_server_connect 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|-----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_SERVER_NETWORK | 网络错误。 |
| (4) HMEC_ERR_SERVER_COMMAND | 命令错误。 |

Remarks

通过 [USER_INFO](#) 结构中的 user_transfer_service 数据项决定是否调用此接口，若用户需要使用中转穿透服务，则调用此接口后，内部会将所有设备的中转穿透策略以及中转穿透服务器相关信息设置妥当，以备以后连接设备时使用。如无特殊配置，总是先尝试直连，失败才会按策略重连。

See Also

[hm_server_connect](#)

hm_server_get_time

获取服务器当前时间。

```
hm_result hm_server_get_time
(
    server_id server,
    uint64* time
)
```

Parameters

server	[in]	hm_server_connect 之返回句柄。
time	[out]	服务器时间，距离 1900 年 1 月 1 日逝去的秒数。

Return Values

成功返回 0，否则返回非 0 值。

See Also

[hm_server_connect](#)

hm_server_get_node_type

获取节点类型。

```
hm_result hm_server_get_node_type  
(  
    node_handle node,  
    NODE\_TYPE\_INFO* tinfo  
)
```

Parameters

node	[in]	设备列表树形结构中某一个节点句柄。
tinfo	[out]	节点类型。

Return Value

成功返回 0，否则返回非 0 值。

Remarks

节点的类型有 4 种：设备、分组、DVS 和通道，通过此接口可以确定此节点是哪种类型，以便后续处理。

hm_server_get_device_url

获取节点 URL。

```
hm_result hm_server_get_device_url  
  
(  
  
    node_handle device,  
  
    cpchar* url  
  
)
```

Parameter

device	[in]	设备列表树形结构中某一个节点句柄。
url	[out]	节点 URL 字符串。

Return Values

成功返回 0，否则返回非 0 值。

Remarks

此接口返回节点 url 的引用，无需释放。

hm_server_get_node_name

获取设备名称。

```
hm_result hm_server_get_node_name  
  
(  
  
    node_handle device,  
  
    cpchar* name  
  
)
```

Parameters

device	[in]	设备列表树形结构中某一个节点句柄。
name	[out]	设备名称。

Return Values

成功返回 0，否则返回非 0 值。

Remarks

此接口返回节点名称的引用，无需释放。

hm_server_get_last_update

获取设备最后在平台的更新时间。

```
hm_result hm_server_get_last_update
(
    node_handle device,
    uint64* time
)
```

Parameters

device	[in]	设备列表树形结构中某一个节点句柄。
time	[out]	设备最后更新时间，此值是距 1900 年 1 月 1 日逝去的秒数。

Return Values

成功返回 0，否则返回非 0 值。

hm_server_get_device_sn

获取设备 SN 号。

```
hm_result hm_server_get_device_sn  
(  
    node_handle device,  
    cpchar* sn  
)
```

Parameters

device	[in]	设备列表树形结构中某一个节点句柄。
sn	[out]	设备 SN 号。

Return Values

成功返回 0，否则返回非 0 值。

hm_server_get_channel_info

获取通道信息。

```
hm_result hm_server_get_channel_info  
(  
    node_handle channel,  
    P\_CHANNEL\_INFO cinfo
```

```
)
```

Parameters

device	[in]	设备列表树形结构中某一个节点句柄。
cinfo	[out]	DVS 通道信息。

Return Values

成功返回 0，否则返回非 0 值。

hm_server_bind_device

绑定设备。

```
hm_result hm_server_bind_device
(
    server_id server,
    pchar sn,
    pchar pwd,
    int32 parentid,
    int32* device_id
)
```

Parameters

server	[in]	hm_server_connect 之返回句柄。
sn	[in]	欲绑定设备的 SN 号。
pwd	[in]	欲绑定设备的密码。

parented	[in]	欲将其加入的分组号。
device_id	[out]	绑定设备的 ID 号。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|-----------------------------|---------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数无效。 |
| (3) HMEC_GEN_CREATE_HANDLE | 创建句柄失败。 |
| (4) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (5) HMEC_ERR_SERVER_NETWORK | 网络错误。 |
| (6) HMEC_ERR_SERVER_COMMAND | 命令错误。 |

See Also

[hm_server_connect](#), [hm_server_unbind_device](#)

hm_server_unbind_device

解绑设备。

```
hm_result hm_server_unbind_device
```

```
(  
    server_id server,  
    int32 device_id  
)
```

Paramters

server	[in]	hm_server_connect 之返回句柄。
device_id	[in]	设备 ID。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|-----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_SERVER_NETWORK | 网络错误。 |
| (4) HMEC_ERR_SERVER_COMMAND | 命令错误。 |

See Also

[hm_server connect](#), [hm_server bind device](#)

hm_server_add_group

添加分组。

```
hm_result hm_server_add_group
(
    server_id server,
    pchar groupname,
    pchar comment,
    int32 parentid
)
```

Parameters

server	[in]	hm_server_connect 之返回句柄。
groupname	[in]	新分组名称。
comment	[in]	为新分组添加注释，不关心注释传 0 即可。
parentid	[in]	父分组 ID。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|-----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数无效。 |
| (3) HMEC_ERR_SERVER_NETWORK | 网络错误。 |
| (4) HMEC_ERR_SERVER_COMMAND | 命令错误。 |

Remarks

此接口在指定的分组（由 parentid 指定）下创建一个新分组，若成功可往该分组下添加设备。

See Also

[hm_server_connect](#)

hm_server_change_device_group

改变设备分组。

```
hm_result hm_server_change_device_group
(
    server_id server,
    int32 device_id,
```

```
int32 group_id
)
```

Parameters

server	[in]	hm_server_connect 之返回句柄。
device_id	[in]	设备 ID。
group_id	[in]	加入分组 ID。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数无效。
(3) HMEC_ERR_SERVER_NETWORK	网络错误。
(4) HMEC_ERR_SERVER_COMMAND	命令错误。

See Also

[hm_server_connect](#)

hm_server_get_tree

获取设备树。

```
hm_result hm_server_get_tree
(
    server_id server,
    tree_handle* htree
)
```

```
)
```

Parameters

server	[in]	hm_server_connect 之返回句柄。
htree	[out]	设备树句柄。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。

Remarks

此接口只有在成功获取设备列表之后才有效。

See Also

[hm_server_connect](#), [hm_server_get_device_list](#), [hm_server_release_tree](#)

hm_server_get_root

获取设备树根节点。

```
hm_result hm_server_get_root
(
    tree_handle tree,
    node_handle* node
)
```

Parameters

tree [in] hm_server_get_tree 之返回句柄。
node [out] 设备树根节点句柄。

Return Values

成功返回 0，否则返回非 0 值。

See Also

[hm_server_get_tree](#)

hm_server_find_device_by_sn

通过 SN 号获得设备节点句柄。

```
node_handle hm_server_find_device_by_sn  
(  
    tree_handle tree,  
    pchar sn  
)
```

Parameters

tree [in] hm_server_get_tree 之返回句柄。
sn [in] 设备 SN 号。

Return Values

若成功，返回找到的设备节点句柄，否则返回 0。

See Also

[hm_server_get_tree](#)

hm_server_release_tree

释放设备树。

```
hm_result hm_server_release_tree  
  
(  
  
    tree_handle tree  
  
)
```

Parameters

tree [in] hm_server_get_tree 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口后，树形结构句柄即失效，此时不应再用此参数调用其他设备树操作相关接口。

See Also

[hm_server_get_tree](#)

hm_server_get_children_count

获取子节点个数。

```
hm_result hm_server_get_children_count  
(  
    node_handle device,  
    int32* count  
)
```

Parameters

device	[in]	设备列表树形结构中某一个节点句柄。
count	[out]	节点下子节点个数。

Return Values

成功返回 0，否则返回非 0 值。

See Also

[hm_server_get_tree](#), [hm_server_get_root](#), [hm_server_get_child_at](#)

hm_server_get_child_at

获取指定位置节点句柄。

```
hm_result hm_server_get_child_at
```



```
(
    node_handle device,
    int32 index,
    node_handle* node
)
```

Parameters

device	[in]	设备列表树形结构中某一个节点句柄。
index	[in]	节点序号。
node	[out]	节点句柄。

Return Values

成功返回 0，否则返回非 0 值。

Remarks

通过 hm_get_root, hm_get_children_count 以及 hm_get_child_at 三个接口即能遍历整棵设备树。

See Also

[hm_server_get_tree](#), [hm_server_get_root](#), [hm_server_get_child_count](#)

编解码相关

hm_audio_init

初始化音频解码器。

```
hm_result_handle hm_audio_init
(
    AUDIO\_ENCODE type,
    audio_codec_handle* haudio
)
```

Parameters

type	[in]	音频编解码类型。
haudio	[out]	音频编解码句柄。

Return Values

返回错误码。

可能返回的错误码：

- (1) HMEC_OK 成功。
- (2) HMEC_GEN_PARAM_INVALID 参数错误。
- (3) HMEC_ERR_AUDIO_SPEEX_INIT 音频编解码库初始化失败。

Remarks

type 可选的类型为 PCMS8 或 SPEEX，ARM 保留未用。

See Also

[hm_audio_uninit](#)

hm_audio_decode

解码音频。

```
hm_result hm_audio_decode
(
    audio_codec_handle handle,
    pointer des,
    int32* des_len,
    pointer src,
    int32 src_len,
    int32 sample
)
```

Parameters

- handle [in] hm_audio_codec_init 之返回句柄。
- des [out] 存放解码后音频数据的缓冲区。
- des_len [in/out] 传入 des 缓冲区的大小，若缓冲区容量太小，则返回错误的同时会给 des_len 设置一个新值。
- src [in] 原始数据缓冲区。
- src_len [in] 原始数据长度。
- sample [in] 采样大小。

Return Values

- 返回错误码。
- 可能返回的错误码：
- (1) HMEC_OK 成功。
 - (2) HMEC_GEN_PARAM_INVALID 参数错误。

- | | |
|---------------------------------|------------|
| (3) HMEC_ERR_AUDIO_BUF_LEN | 传出缓冲区长度错误。 |
| (4) HMEC_ERR_AUDIO_FIND_DECODER | 解码类型错误。 |

See Also

[hm_audio_init](#), [hm_audio_encode](#)

hm_audio_encode

编码音频。

```
hm_result hm_audio_encode
```

```
(  
  
    audio_codec_handle handle,  
  
    pointer des,  
  
    int32* des_len,  
  
    pointer src,  
  
    int32 src_len,  
  
    int32 sample  
  
)
```

Parameters

- | | | |
|---------|----------|---|
| handle | [in] | hm_audio_codec_init 之返回句柄。 |
| des | [out] | 存放编码后音频数据的缓冲区。 |
| des_len | [in/out] | 传入 des 缓冲区的大小，若缓冲区容量太小，则返回错误的同时会给 des_len 设置一个新值。 |
| src | [in] | 原始数据缓冲区。 |
| src_len | [in] | 原始数据长度。 |

sample [in] 采样大小。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|---------------------------------|------------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_AUDIO_BUF_LEN | 传出缓冲区长度错误。 |
| (4) HMEC_ERR_AUDIO_FIND_ENCODER | 编码类型错误。 |

See Also

[hm_audio_init](#), [hm_audio_decode](#)

hm_audio_uninit

释放音频解码器。

```
hm_result hm_audio_uninit
(
    audio_codec_handle handle
)
```

Parameters

handle [in] hm_audio_codec_init 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口后，音频编解码句柄即失效，此时不应再用此参数调用其他音频编解码操作相关接口。

See Also

[hm_audio_init](#)

hm_audio_capture_init

初始化音频采集器。

```
hm_result hm_audio_capture_init
(
    audio_capture_handle* handle
)
```

Parameters

handle [in] 返回音频采集句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|-------------|-----|
| (1) HMEC_OK | 成功。 |
|-------------|-----|

- (2) HMEC_GEN_PARAM_INVALID

参数错误。
- (3) HMEC_GEN_CREATE_HANDLE

创建句柄失败。
- (4) HMEC_ERR_AUDIO_INIT

初始化音频采集器失败。
- (5) HMEC_ERR_AUDIO_CAP_ALREADY_INIT

重复初始化音频采集器。

Remarks

此接口目前仅在 Windows 平台上有效，使用此接口前，请确保已经成功安装微软 32 位 DirectX 9.0c 及以上版本库。

See Also

[hm_audio_capture_uninit](#)

hm_audio_capture_set_callback

设置音频采集器回调。

```
hm_result hm_audio_capture_set_callback
(
    audio_capture_handle handle,
    cb_audio_capture cb,
    user_data data
)
```

Parameters

- handle

[in]

hm_audio_data_capture_init 接口返回之句柄。
- cb

[in]

用户注册回调。

函数签名式：

void (user_data data, hm_result result, pointer buf, int32 len)

data	[out]	用户数据，和此接口传入的 data 参数相同。
result	[out]	错误码。
pointer	[out]	原始音频数据缓冲区。
len	[out]	音频数据缓冲区长度。

data [in] 用户数据。

Return Values

返回错误码。

可能返回的错误码：

- (1) HMEC_OK 成功。
- (2) HMEC_GEN_PARAM_INVALID 参数错误。

See Also

[hm_audio_capture_init](#), [hm_audio_capture_start](#), [hm_audio_capture_stop](#)

hm_audio_capture_start

开启音频采集。

hm_result hm_audio_capture_start

(

audio_capture_handle handle
)

Parameter

handle [in] hm_audio_data_capture_init 接口返回之句柄。

Return Values

返回错误码。

可能的错误码：

- (1) HMEC_OK成功。
- (2) HMEC_GEN_PARAM_INVALID参数错误。

Remarks

调用此接口前，需先调用 `hm_audio_data_capture_set_callback`，将音频数据回调设置妥当。

See Also

[hm_audio_capture_init](#), [hm_audio_capture_stop](#)

hm_audio_capture_stop

关闭音频采集。

```
hm_result hm_audio_capture_stop
(
    audio_capture_handle handle
)
```

Parameter

handle [in] `hm_audio_data_capture_init` 接口返回之句柄。

Return Values

返回错误码。

可能的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

此接口只是单纯停止数据回调，若要释放 handle，请调用 hm_audio_data_capture_uninit 接口。

See Also

[hm_audio_capture_init](#), [hm_audio_capture_start](#)

hm_audio_capture_uninit

释放音频采集器。

```
hm_result hm_audio_capture_uninit
(
    audio_capture_handle handle
)
```

Parameter

handle	[in]	hm_audio_data_capture_init 接口返回之句柄。
--------	------	-------------------------------------

Return Values

返回错误码。

可能的错误码：

- | | |
|-------------|-----|
| (1) HMEC_OK | 成功。 |
|-------------|-----|

(2) HMEC_GEN_PARAM_INVALID 参数错误。

Remarks

此接口会释放 handle，此时不应再用此 handle 调用其他采集接口。

See Also

[hm_audio_capture_init](#)

hm_audio_player_init

初始化音频播放器。

```
hm_result hm_audio_player_init
(
    audio_player_handle* handle,
    window_handle wnd,
    AUDIO\_ENCODE audio_type,
    int32 sample,
    int32 chn_num,
    int32 bitwidth
)
```

Parameter

handle	[in/out]	返回音频播放器句柄。
audio_type	[in]	音频类型。
sample	[in]	音频采样率。

chn_num	[in]	音频通道数。
bitwidth	[in]	音频位宽。

Return Values

返回错误码。

可能的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_AUDIO_PLAYER_CREATE	创建播放器失败。
(4) HMEC_AUDIO_PLAYER_ALREADY_INIT	重复初始化。
(5) HMEC_AUDIO_PLAYER_UNKNOWN_TYPE	未知音频类型。
(6) HMEC_AUDIO_PLAYER_SET_COOP_LEVEL	设置协作层失败。
(7) HMEC_AUDIO_PLAYER_CREATE_BUF	分配存储空间失败。

Remarks

若调用成功，handle 会被赋值音频播放器句柄，此后的操作均需使用此句柄。建议使用默认参数初始化播放器。

See Also

[hm_audio_player_uninit](#)

hm_audio_player_start

开启音频播放。

hm_result hm_audio_player_start

(

audio_player_handle handle

```
)
```

Parameter

handle	[in]	hm_audio_player_init 接口返回之句柄。
--------	------	-------------------------------

Return Values

返回错误码。

可能的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口播放器即开始接收数据，此时可不断调用 hm_audio_player_insert_frame 向播放器传送数据。

See Also

[hm_audio_player_init](#), [hm_audio_player_stop](#)

hm_audio_player_stop

关闭音频播放。

```
hm_result hm_audio_player_stop
(
    audio_player_handle handle
)
```

Parameter

handle	[in]	hm_audio_player_init 接口返回之句柄。
--------	------	-------------------------------

Return Values

返回错误码。

可能的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口播放器将停止接收音频数据，此时不应再调用hm_audio_player_insert_frame 向播放器传送数据。

See Also

[hm_audio_player_init](#), [hm_audio_player_start](#)

hm_audio_player_insert_frame

插入音频帧。

```
hm_result hm_audio_player_insert_frame
(
    audio_player_handle handle,
    cpchar buf,
    int32 len
)
```

Parameter

handle [in] hm_audio_player_init 接口返回之句柄。

Return Values

返回错误码。

可能的错误码：

- (1) HMEC_OK 成功。
- (2) HMEC_GEN_PARAM_INVALID 参数错误。

Remarks

调用此接口向播放器传送待播放音频数据，可在接收音频数据回调当中调用。

See Also

[hm_audio_player_init](#)

hm_audio_player_uninit

释放音频播放器。

```
hm_result hm_audio_player_uninit
(
    audio_player_handle handle
)
```

Parameter

handle [in] hm_audio_player_init 接口返回之句柄。

Return Values

返回错误码。

可能的错误码:

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口后，音频播放器句柄即失效，此时不应再用此句柄调用其他音频播放操作。

See Also

hm_audio_player_init

hm_video_init

初始化视频解码器。

```
hm_result hm_video_init
(
    VIDEO_ENCODE type,
    video_codec_handle* hvideo
)
```

Parameters

type	[in]	视频解码类型。
hvideo	[in]	返回视频解码句柄。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_GEN_CREATE_HANDLE	创建句柄失败。
(4) HMEC_ERR_VIDEO_FIND_DECODER	未知解码类型。
(5) HMEC_ERR_VIDEO_ALLOC_CONTEXT	分配上下文失败。
(6) HMEC_ERR_VIDEO_CODEC_OPEN	打开解码器失败。
(7) HMEC_ERR_VIDEO_ALLOC_FRAME	分配存储空间失败。

See Also

[hm_video_uninit](#)

hm_video_decode_yuv

解码一帧 YUV 数据(支持解码 H264 和 H265)。

```
hm_result hm_video_decode_yuv
```

```
(
    video_codec_handle handle,
    pointer video_data,
    int32 video_len,
    yuv_handle* yuv
)
```

Parameters

handle	[in]	hm_video_codec_init 之返回句柄。
video_data	[in]	原始视频数据缓冲区。
video_len	[in]	原始视频数据长度。
yuv	[out]	解码后数据容器句柄。

Return Values

成功返回 0，否则返回非 0 值。

Remarks

此接口解码后的数据格式为 yuv 4:2:0，支持 yuv 颜色空间的平台调用此接口的效率较高；在不支持 yuv 颜色空间的平台上请调用 hm_video_decode_bitmap 代替之，将数据转换为给定的 RGB 格式数据。

返回的句柄使用完毕后，请调用 hm_video_release_yuv 加以释放。

注意：实时视频和远程录像回放时，回调出的 P_FRAME_DATA 中 frame_info 的 frame_type 字段意义为：0-H264P 帧，1-H264I 帧，0x20-H265P 帧，0x21-H265I 帧；本地录像回调出的 P_PLAYBACK_FRAME 中的 ftype 字段意义同实时视频。

See Also

[hm_video_init](#), [hm_video_decode_bitmap](#), [hm_video_release_yuv](#)

hm_video_decode_bitmap

解码一帧 RGB 数据。

hm_result hm_video_decode_bitmap

(

```

video_codec_handle handle,

pointer video_data,

int32 video_len,

BITMAP\_FORMAT bf,

bitmap_handle* bmp

)

```

Parameters

handle	[in]	hm_video_codec_init 之返回句柄。
video_data	[in]	原始视频数据缓冲区。
video_len	[in]	原始视频数据长度。
bf	[in]	rgb 格式，目前支持的格式为 EBF_BGR32, EBF_RGB32, EBF_RGBA32, EBF_RGB16_565。
bmp	[out]	解码后数据容器句柄。

Return Values

成功返回 0，否则返回非 0 值。发生错误的可能是解码失败，或是传入的 rgb 格式非法。

Remarks

在支持 yuv 颜色空间的平台上请优先考虑调用 hm_video_decode_yuv 以提高效率。返回的句柄使用完毕后，请调用 hm_video_release_bitmap 加以释放。

See Also

[hm_video_init](#), [hm_video_decode_yuv](#), [hm_video_release_bitmap](#)

hm_video_release_yuv

释放解码后容器句柄。

```
hm_result hm_video_release_yuv  
(  
    yuv_handle handle  
)
```

Parameters

handle [in] hm_video_decode_yuv 之返回值。

Return Values

返回错误码。

可能返回的错误码：

- (1) HMEC_OK 成功。
- (2) HMEC_GEN_PARAM_INVALID 参数错误。

Remarks

此接口会清空解码后的 yuv 数据，此时不应再使用此 yuv 句柄。

See Also

[hm_video_decode_yuv](#)

hm_video_release_bitmap

释放解码后容器句柄。

```
hm_result hm_video_release_bitmap
```

```
(
    bitmap_handle handle
)
```

Parameters

handle [in] hm_video_decode_bitmap 之返回值。

Return Values

返回错误码。

可能返回的错误码：

- (1) HMEC_OK 成功。
- (2) HMEC_GEN_PARAM_INVALID 参数错误。

Remarks

此接口会清空解码后的 bitmap 数据，此时不应再使用此 bitmap 句柄。

See Also

[hm_video_decode_bitmap](#)

```
hm_video_yuv_2_rgb
```

YUV 数据转换成 RGB 数据。

```
hm_result hm_video_yuv_2_rgb
```

```
(
    yuv_handle handle,
    BITMAP\_FORMAT bf,
    bitmap_handle* bmp
)
```

Parameters

handle	[in]	hm_video_decode_yuv 之返回值。
bf	[in]	转换出的 RGB 格式。
bmp	[out]	转换后数据容器句柄。

Return Values

成功返回 0，否则返回非 0 值。

Remarks

调用此接口将 yuv 句柄转换为 bitmap 句柄，此句柄中包含相应的 rgb 数据。

See Also

[hm_video_get_bitmap_data](#), [hm_video_get_bitmap_file_data](#), [hm_video_get_bitmap_info](#)

hm_video_get_yuv_data

获取 YUV 数据信息结构。

hm_result hm_video_get_yuv_data

```
(
    yuv_handle handle,
    P\_YUV\_PICTURE yuv_pic,
)
```

Parameters

handle	[in]	hm_video_decode_yuv 之返回值。
yuv_pic	[in/out]	传出 YUV_PICTURE 结构数据，由用户事先分配空间。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

See Also

[hm_video_decode_yuv](#)

hm_video_get_bitmap_data

获取 RGB 数据原始指针。

```
hm_result hm_video_get_bitmap_data
(
    bitmap_handle handle,
    uint32* len,
```

```
cpchar* data
```

```
)
```

Parameters

handle	[in]	hm_video_decode_bitmap 之返回值。
len	[in/out]	数据长度。
data	[out]	数据原始指针。

Return Values

成功返回 0，否则返回非 0 值。

See Also

[hm_video_decode_bitmap](#)

hm_video_get_bitmap_file_data

获取 RGB 文件数据，可直接存为 BMP 格式文件。

```
hm_result hm_video_get_bitmap_file_data
```

```
(
```

```
    bitmap_handle handle,
```

```
    uint32* len,
```

```
    cpchar* data
```

```
)
```

Parameters

handle	[in]	hm_video_decode_bitmap 之返回值。
len	[in/out]	数据长度。
data	[out]	原始文件数据指针。

Return Values

成功返回 0，否则返回非 0 值。

See Also

[hm_video_decode_bitmap](#), [hm_video_yuv_2_bitmap](#)

hm_video_get_bitmap_info

获取 RGB 数据信息。

```
hm_result hm_video_get_bitmap_info
(
    bitmap_handle handle,
    P\_BITMAP\_INFO\_HEADER hinfo
)
```

Parameters

handle	[in]	hm_video_decode_bitmap 之返回值。
hinfo	[out]	RGB 数据信息。

Return Values

成功返回 0，否则返回非 0 值。

See Also

[hm_video_decode_bitmap](#)

hm_video_reset

重置解码器（不释放）。主要用于视频断线后，通知解码器等待第一个 I 帧才开始解码。

```
hm_result hm_video_reset  
  
(  
  
    video_codec_handle handle  
  
)
```

Parameters

handle [in] hm_video_codec_init 之返回句柄。

hm_video_uninit

释放视频解码器。

```
hm_result hm_video_uninit  
  
(  
  
    video_codec_handle handle  
  
)
```

Parameters

handle [in] hm_video_codec_init 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- (1) HMEC_OK
- 成功。
- (2) HMEC_GEN_PARAM_INVALID
- 参数错误。

Remarks

调用此函数后，video_codec_handle 句柄即被释放，此时不应再用此值调用其他视频解码接口。

See Also

[hm_video_init](#)

视频显示相关

hm_video_display_open_port

初始化视频显示端口。

```
hm_result hm_video_display_open_port
(
    window_handle handle,
    port_handle* port
)
```

Parameters

- handle
- [in]
- 欲显示画面窗口的句柄。
- port
- [in/out]
- 显示端口值。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|--------------------------------|-----------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_GEN_VIDEO_PLAYER_INIT | 播放器初始化失败。 |

Remarks

调用此接口，正确传入窗口句柄值，若函数执行成功，port 会被赋值显示端口值，调用视频显示的其他接口均需使用此值。

See Also

[hm_video_display_close_port](#)

hm_video_display_start

开启视频显示。

```
hm_result hm_video_display_start
```

```
(
```

```
    port_handle port,
```

```
    int32 vw,
```

```
    int32 vh,
```

```
    int32 fps
```

```
)
```

Parameters

port	[in]	hm_video_display_open_port 之返回端口值。
vw	[in]	视频宽。
vh	[in]	视频高。
fps	[in]	视频帧率。

Return Values

返回错误码。

可能返回的错误码：

- (1) HMEC_OK

成功。
- (2) HMEC_GEN_PARAM_INVALID

参数错误。

Remarks

调用此接口，使得播放器开始准备接收数据，之后可连续调用hm_video_display_input_data给播放器传送数据。若不关心视频宽高和帧率等信息，可以使用默认值调用接口。

See Also

[hm_video_display_stop](#)

hm_video_display_input_data

向显示端口传入视频数据。

```
hm_result hm_video_display_input_data
(
    port_handle port,
    cpointer video_data,
```

```
int32 len
)
```

Parameters

port	[in]	hm_video_display_open_port 之返回端口值。
video_data	[in]	原始视频数据。
len	[in]	原始视频数据长度。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口，向播放器传送待播放的视频数据，当 hm_video_display_stop 调用后，此接口变得无效，直到下次调用 hm_video_display_start 此接口即恢复正常。

See Also

[hm_video_display_start](#), [hm_video_display_stop](#)

hm_video_display_get_buffer_size

获取当前显示队列大小。

```
int32 hm_video_display_get_buffer_size
(
```

```
port_handle port
)
```

Parameters

port [in] hm_video_display_open_port 之返回端口值。

Return Values

返回当前缓冲区大小。

Remarks

调用此接口以得知当前使用的缓冲区大小，控制传送数据的速率。

hm_video_display_stop

关闭视频显示。

```
hm_result hm_video_display_stop
(
    port_handle port
)
```

Parameters

port [in] hm_video_display_open_port 之返回端口值。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口，播放器将停止接收数据，此时 `hm_video_display_input_data` 会变得无效。

See Also

[hm_video_display_start](#)

hm_video_display_close_port

释放视频显示端口。

```
hm_result hm_video_display_close_port
(
    port_handle port
)
```

Parameters

`port` [in] `hm_video_display_open_port` 之返回端口值。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口，端口值变得无效，此时不应再使用此端口值调用其他视频显示操作。

See Also

[hm_video_display_open_port](#)

本地录像/回放相关

hm_util_local_record_init

初始化本地录像。

```
hm_result hm_util_local_record_init
(
    P\_LOCAL\_RECORD\_PARAM param,
    local_record_handle* hrecord
)
```

Parameters

param	[in]	录像相关参数。
hrecord	[in/out]	返回录像句柄。

Return Values

返回错误码。
可能返回的错误码：

(1) HMEC_OK	成功。
-------------	-----

- | | |
|-----------------------------------|---------|
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_UTIL_INIT | 初始化失败。 |
| (4) HMEC_ERR_UTIL_FILE_WRITE_FAIL | 写文件失败。 |
| (5) HMEC_ERR_UTIL_FILE_CREATE | 创建文件失败。 |

Remarks

成功调用此接口，hrecord 将被赋值录像句柄，录像的后续操作均需使用此句柄。

See Also

[hm_util_local_record_uninit](#)

hm_util_local_record_write

向录像文件写入帧数据。

```
hm_result hm_util_local_record_write
(
    local_record_handle hrecord,
    P_FRAME_DATA frame,
    uint32* time_cost
)
```

Parameters

- | | | |
|-----------|----------|----------------------------------|
| hrecord | [in] | hm_util_local_record_init 之返回句柄。 |
| frame | [in] | 帧信息。 |
| time_cost | [in/out] | 传出当前已录像时长。 |

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_ERR_UTIL_INIT	初始化失败。
(4) HMEC_ERR_UTIL_FILE_CREATE	创建录像文件失败。
(5) HMEC_ERR_UTIL_FILE_NOT_OPEN	打开文件失败。
(6) HMEC_ERR_UTIL_FILE_WRITE_FAIL	写文件失败。

Remarks

循环调用此接口可产生录像文件，可根据 time_cost 返回的录像时长决定是否停止录像，欲停止录像请务必调用 hm_util_local_record_uninit，此时会在给定的路径下生成一个*.hmv 格式录像文件。

See Also

[hm_util_local_record_init](#)

hm_util_local_record_uninit

释放本地录像对象。

```
hm_result hm_util_local_record_uninit
(
    local_record_handle hrecord
)
```

Parameters

hrecord [in] hm_util_local_record_init 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口停止录像，并释放录像句柄，此时不应再进行录像相关操作。

See Also

[hm_util_local_record_init](#)

hm_util_local_playback_init

初始化本地回放。

```
hm_result hm_util_local_playback_init
(
    P\_LOCAL\_RECORD\_PARAM param,
    local_playback_handle* hplayback,
    PB\_MODE pbm
)
```

Parameters

param [in/out] 用户申请的录像信息结构。

hplayback	[in]	返回回放句柄。
pbm	[in]	回放模式。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------------|---------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_UTIL_FILE_INVALID | 录像文件无效。 |
| (4) HMEC_ERR_UTIL_FILE_NOT_OPEN | 打开文件失败。 |
| (5) HMEC_ERR_UTIL_FILE_READ_FAIL | 读取文件失败。 |

Remarks

成功调用此接口，hplayback 会被赋值回放句柄，此后的回放相关操作均需使用此句柄。

目前回放模式有两种：1) 主动模式。2) 被动模式。

对于主动模式：回放的基本步骤是：1) 初始化回放句柄。2) 调用 hm_util_local_playback_get_one_frame 主动获取一帧数据，直到文件尾。3) 可以调用 hm_util_local_playback_get_position, hm_util_local_playback_set_position 接口，其他接口对于主动模式无效。4) 释放回放句柄。

对于被动模式：回放的基本步骤是：1) 初始化回放句柄。2) 设置回放数据回调。3) 开始回放。4) 一些回放相关操作，如暂停，跳帧等。5) 停止回放。6) 释放回放句柄。

回放接口仅支持*.hmv 格式录像文件。

See Also

[hm_util_local_playback_uninit](#)

hm_util_local_playback_get_one_frame

主动模式获取一帧数据。

```
hm_result hm_util_local_playback_get_one_frame
(
    local_playback_handle hplayback,
    P_PLAYBACK_FRAME frame
)
```

Parameters

hplayback	[in]	hm_util_local_playback_init 之返回句柄。
frame	[out]	帧数据信息。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。
(3) HMEC_ERR_UTIL_END_OF_FILE	文件已达末尾。

Remarks

调用此接口返回一帧数据，数据帧根据时间戳依次返回。接口返回 HMEC_ERR_UTIL_END_OF_FILE 表明文件已读取完毕。

hm_util_local_playback_set_callback

设置录像数据回调函数。

```
hm_result hm_util_local_playback_set_callback
(
    local_playback_handle hplayback,
    cb_util_playback cb,
    user_data data
)
```

Parameters

hplayback	[in]	hm_util_local_playback_init 之返回句柄。
cb	[in]	接收回放数据回调。
data	[in]	用户数据。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

此接口用于设置开始回放时回放数据回调，请确保在开始回放之前已正确设置回调，否则部分数据将无法播放。

cb_util_playback 回调函数签名：

```
void LocalPlaybackCallback(user_data data, P\_PLAYBACK\_FRAME frame)
{
    if (frame == NULL) return;
    switch (frame->ftype)
    {
```

```
case HME_RFT_P:
case HME_RFT_I:
case HME_RFT_H265_P:
case HME_RFT_H265_I:
    // TODO 拷贝视频数据，加入视频帧队列，等待解码
    break;
case HME_RFT_AUDIO:
    // TODO 拷贝音频数据，加入音频帧队列，等待解码
    break;
}
}
```

hm_util_local_playback_start

开启回放。

```
hm_result hm_util_local_playback_start
(
    local_playback_handle hplayback
)
```

Parameters

hplayback [in] hm_util_local_playback_init 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口开始接收回放数据，请确保在调用此接口之前已正确设置回调。

See Also

[hm_util_local_playback_stop](#)

hm_util_local_playback_stop

停止回放。

```
hm_result hm_util_local_playback_stop  
(  
    local_playback_handle hplayback  
)
```

Parameters

hplayback [in] hm_util_local_playback_init 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口停止视频回放，当再次开启回放时将会从开始处开始回放。

See Also

[hm_util_local_playback_start](#)

hm_util_local_playback_pause

暂停回放。

```
hm_result hm_util_local_playback_pause  
(  
    local_playback_handle hplayback  
)
```

Parameters

hplayback [in] hm_util_local_playback_init 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口暂停回放，要继续回放请调用 hm_util_local_playback_resume，视频将从暂停处恢复。

See Also

[hm_util_local_playback_resume](#)

hm_util_local_playback_resume

恢复回放。

```
hm_result hm_util_local_playback_resume  
(  
    local_playback_handle hplayback  
)
```

Parameters

hplayback [in] hm_util_local_playback_init 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口恢复视频回放，同 hm_util_local_playback_pause 配合使用。

See Also

[hm_util_local_playback_pause](#)

hm_util_local_playback_get_rate

获取当前回放速率。

```
hm_result hm_util_local_playback_get_rate
```

```
(
    local_playback_handle hplayback,
    PLAYBACK\_RATE* rate
)
```

Parameters

hplayback	[in]	hm_util_local_playback_init 之返回句柄。
rate	[out]	当前回放速率。

Return Values

成功返回 0，否则返回非 0 值。

Remarks

调用此接口获取当前回放速率，支持的回放速率为：原始速率（1 倍速），2 倍速，4 倍速，8 倍速。

See Also

[hm_util_local_playback_set_rate](#)

hm_util_local_playback_set_rate

设置回放速率。

```
hm_result hm_util_local_playback_set_rate
(
    local_playback_handle hplayback,
```

```
PLAYBACK_RATE rate
)

```

Parameters

hplayback	[in]	hm_util_local_playback_init 之返回句柄。
rate	[in]	回放速率。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。

Remarks

调用此接口设置回放速率，支持的回放速率为：原始速率（1 倍速），2 倍速，4 倍速，8 倍速。

See Also

[hm_util_local_playback_get_rate](#)

hm_util_local_playback_get_position

获取当前回放位置百分比，单位 100。

```
hm_result hm_util_local_playback_get_position
(
    local_playback_handle hplayback,

```

```
real64* pos
```

```
)
```

Parameters

hplayback [in] hm_util_local_playback_init 之返回句柄。

Return Values

当前回放位置百分比。

Remarks

调用此接口获取当前帧时间戳与整个播放时长的百分比,上层使用此百分比可以设置回放滚动条等界面相关控件。

See Also

[hm_util_local_playback_set_position](#)

hm_util_local_playback_set_position

设置回放位置。

```
hm_result hm_util_local_playback_set_position
```

```
(
```

```
    local_playback_handle hplayback,
```

```
    real64 pos
```

```
)
```

Parameters

hplayback	[in]	hm_util_local_playback_init 之返回句柄。
pos	[in]	回放位置百分比。

Return Values

返回错误码。

可能返回的错误码：

(1) HMEC_OK	成功。
(2) HMEC_GEN_PARAM_INVALID	参数错误。

Remarks

调用此接口设置回放位置百分比，一般由上层回放滚动条计算得出，传进接口中。

See Also

[hm_util_local_playback_get_position](#)

hm_util_local_playback_step_forward

单帧向前回放。

```
hm_result hm_util_local_playback_step_forward
(
    local_playback_handle hplayback
)
```

Parameters

hplayback	[in]	hm_util_local_playback_init 之返回句柄。
-----------	------	------------------------------------

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口将向后跳过一帧。

See Also

[hm_util_local_playback_step_backward](#)

hm_util_local_playback_step_backward

单帧向后回放。

```
hm_result hm_util_local_playback_step_backward  
  
(  
  
    local_playback_handle hplayback  
  
)
```

Parameters

hplayback [in] hm_util_local_playback_init 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口将向前跳过一帧。

See Also

[hm_util_local_playback_step_forward](#)

hm_util_local_playback_uninit

释放录像回放对象。

```
hm_result hm_util_local_playback_uninit
(
    local_playback_handle hplayback
)
```

Parameters

hplayback [in] hm_util_local_playback_init 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口释放回放句柄，此时不应再使用此句柄进行回放相关操作。

See Also

[hm_util_local_playback_init](#)

hm_util_local_capture

本地抓图。

```
hm_result hm_util_local_capture
(
    cpchar path,
    yuv_handle handle,
    BITMAP\_FORMAT bf
)
```

Parameters

path	[in]	抓图保存路径，需符合相应平台路径要求。
handle	[in]	hm_video_decode_yuv 之返回句柄。
bf	[in]	位图格式。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

- (3) HMEC_ERR_UTIL_FILE_CREATE 创建文件失败。
- (4) HMEC_ERR_UTIL_FILE_WRITE_FAIL 写入文件失败。

Remarks

给出图片文件保存地址，此接口会在给定路径下生成给定格式的位图文件。

局域网搜索相关

hm_util_lan_device_search_init

初始化局域网搜索。

```
hm_result hm_util_lan_device_search_init
(
    P\_LAN\_DEVICE\_SEARCH\_PARAM param,
    lan_device_search_handle* handle
)
```

Parameters

- | | | |
|--------|----------|--------------------------|
| param | [in] | 搜索信息结构，设置搜索模式，数据回调和用户数据。 |
| handle | [in/out] | 返回局域网搜索句柄。 |

Return Values

返回错误码。
可能返回的错误码：

- | | |
|----------------------------|--------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_UTIL_INIT | 初始化失败。 |

Remarks

调用此接口初始化局域网搜索句柄，此后的局域网搜索相关操作均需使用此句柄。

See Also

[hm_util_lan_device_search_uninit](#)

hm_util_lan_device_search_query

开启局域网搜索。

```
hm_result hm_util_lan_device_search_query
(
    lan_device_search_handle handle
)
```

Parameters

handle [in/out] hm_util_lan_device_search_init 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|--------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |
| (3) HMEC_ERR_UTIL_INIT | 初始化失败。 |

Remarks

调用此接口开启局域网搜索，请确保在此之前已正确设置数据回调。

See Also

[hm_util_lan_device_search_init](#)

hm_util_lan_device_search_uninit

释放局域网搜索对象。

```
hm_result hm_util_lan_device_search_uninit
(
    lan_device_search_handle handle
)
```

Parameters

handle [in/out] hm_util_lan_device_search_init 之返回句柄。

Return Values

返回错误码。

可能返回的错误码：

- | | |
|----------------------------|-------|
| (1) HMEC_OK | 成功。 |
| (2) HMEC_GEN_PARAM_INVALID | 参数错误。 |

Remarks

调用此接口停止局域网搜索并释放搜索句柄，此时不应再使用此句柄进行局域网搜索相关操作。

See Also

[hm_util_lan_device_search_init](#)

布撤防

hm_pu_arming_area

布防。

```
hm_result hm_pu_arming_area
(
    user_id id,
    int32 area_id,
    cpchar expand
)
```

Parameters

id	[in]	hm_pu_login 或 hm_pu_login_ex 之返回句柄。
area_id	[in]	保留字段，暂未启用
expand	[in]	保留字段，暂未启用

Return Values

成功返回 HMEC_OK

Remarks

hm_pu_disarming_area

撤防。

```
hm_result hm_pu_disarming_area
(
    user_id id,
    int32 area_id,
    cpchar expand
)
```

Parameters

- id [in] [hm_pu_login](#) 或 [hm_pu_login_ex](#) 之返回句柄。
- area_id [in] 保留字段，暂未启用
- expand [in] 保留字段，暂未启用

Return Values

成功返回 HMEC_OK

Remarks

结构体描述

FRAME_INFO

字段	类型	含义
channel	uint16	通道号 0、1、2...
data_type	uint16	视频类型，见 VIDEO_STREAM
stream_type	uint16	码流类型，见 CODE_STREAM

frame_type	uint16	帧类型，见 RAW_FRAME_TYPE
absolute_timestamp	uint64	时间戳，单位毫秒

FRAME_DATA

字段	类型	含义
frame_info	FRAME_INFO	帧信息
frame_len	uint32	帧长度
frame_stream	cpointer	帧数据

CHANNEL_CAPACITY

字段	类型	含义
channel_name	char[]	通道名称
video_name	char[]	视频名称
video_support	bool	是否支持视频
audio_support	bool	是否支持音频
talk_support	bool	是否支持对讲
ptz_support	bool	是否支持云台
audio_code_type	AUDIO_ENCODE	音频编码类型
audio_sample	int32	音频采样率
audio_channel	int32	声道数

DEVICE_INFO

字段	类型	含义
dev_name	char[]	设备名称
dev_type	char[]	设备类型
sn	char[]	设备 SN 号
total_channel	int32	通道总数
alarm_in_count	int32	报警输入通道数

alarm_out_count	int32	报警输出通道数
sensor_count	int32	传感器数量
support_third_stream	int32	是否支持三码流
channel_capacity	P_CHANNEL_CAPACITY []	DVS 通道能力

OPEN_VIDEO_PARAM

字段	类型	含义
channel	uint32	打开视频的通道号
cs_type	CODE_STREAM	码流类型
vs_type	VIDEO_STREAM	视频类型
cb_data	cb_pu_data	视频回调函数
wnd	window_handle	窗口句柄（仅 WINDOWS 平台）
data	user_data	回调用户数据

cb_pu_data 签名：

```
void Callback(user_data data, P\_FRAME\_DATA frame, hm_result result)
{
    if (result == HMEC_OK && frame)
    {
        if (frame->frame_info.frame_type == HME_RFT_P ||
            frame->frame_info.frame_type == HME_RFT_I ||
            frame->frame_info.frame_type == HME_RFT_H265_P ||
            frame->frame_info.frame_type == HME_RFT_H265_I)
        {
            // TODO 视频数据拷贝，加入视频帧队列
        }
        else if (frame->frame_info.frame_type == HME_RFT_INFO)
        {
            // TODO Nothing
        }
        else
        {
            // TODO 音频数据拷贝，加入音频帧队列
        }
    }
}
```

OPEN_VIDEO_RES

字段	类型	含义
channel	uint32	打开视频的通道号
video_type	VIDEO_ENCODE	视频编码类型
image_width	int32	视频画面宽
image_height	int32	视频画面高
fps	int32	视频帧率
basetime	int64	视频基准时间

OPEN_AUDIO_PARAM

字段	类型	含义
channel	uint32	打开音频的通道号
cb_data	cb_pu_data	音频回调函数
data	user_data	回调用户数据

cb_pu_data 回调签名见 [OPEN_VIDEO_PARAM](#)

OPEN_AUDIO_RES

字段	类型	含义
channel	uint32	打开音频的通道号
audio_type	AUDIO_ENCODE	音频编码类型
sample	int32	音频采样率
audio_channel	int32	音频声道数

OPEN_TALK_PARAM

字段	类型	含义
channel	uint32	打开音频的通道号
audio_type	AUDIO_ENCODE	音频编码类型
sample	int32	音频采样率
audio_channel	int32	音频声道数

REMOTE_CAPTURE_PIC_PARAM

字段	类型	含义
channel	uint32	抓图通道号

ALARM_INFO

字段	类型	含义
key	char	此条报警 ID
sn	char	设备 SN 号
device_name	char	设备名称
channel	int32	通道号
area_id	int32	防区 ID
alarm_dev_type	ALARM_DEVICE_TYPE	报警设备类型
alarm_type	ALARM_TYPE	报警类型
happened_time	char	报警发生时间
context	char	用户自定义防区名称

OPEN_ALARM_PARAM

字段	类型	含义
cb_alarm	cb_pu_alarm	报警回调
data	user_data	用户回调数据

cb_pu_alarm 回调签名:

```
void AlarmCallback(user_data data, P\_ALARM\_INFO info, hm_result result)
{
    // TODO
}
```

WIFI_INFO

字段	类型	含义
encrypt	uint8	是否加密
key_type	uint8	密码类型（0 代表 16 进制，1 代表 ASCII）
encrypt_type	uint8	加密类型（0 代表 TKIP，1 代表 AES）

mode	uint8	安全模式（0 代表 OPEN，1 代表 WPA_PSK，2 代表 WPA2_PSK）
ssid	char	WIFI 广播名称

FIND_FILE_PARAM

字段	类型	含义
channel	uint32	通道号
record_type	VIDEO_RECORD	录像类型
search_mode	SEARCH_MODE	查询方式
start_time	char[]	开始时间，时间格式形如“2009-09-29 13:50:32”
end_time	char[]	结束时间，格式同上
search_month	char[]	查询月份, 1-12

FIND_FILE_DATA

字段	类型	含义
start_time	char[]	录像开始时间
end_time	char[]	录像结束时间
file_name	char[]	文件名
record_type	VIDEO_RECORD	录像类型

DELETE_RECORD_FILE_PARAM

字段	类型	含义
channel	int32	通道号
file_name	char[]	文件名

PLAYBACK_PARAM

字段	类型	含义
----	----	----

channel	int32	通道号
play_mode	REMOTE_PLAYBACK_MODE	
file_name	char[]	文件名
play_time	uint64	播放位置偏移
start_time	uint64	开始时间
end_time	uint64	结束时间
key	char[]	报警 key
frame_type	int32	
cb_data	cb_pu_playback_data	回放数据回调
wnd	window_handle	窗口句柄（仅 widnows 平台）
data	user_data	用户回调数据

cb_pu_playback_data 回调签名：

```
void RemotePlaybackCallback(user_data data, bool over, P_FRAME_DATA frame,
hm_result result)
{
    if (frame == NULL) return;
    if (frame->frame_len == 0) return; // 视频结束
    switch(frame->frame_info.frame_type)
    {
        case HME_RFT_P:
        case HME_RFT_I:
        case HME_RFT_H265_P:
        case HME_RFT_H265_I:
            // TODO 拷贝视频数据，加入视频帧队列，等待解码
            break;
        case HME_RFT_PCM:
        case HME_RFT_SPEEX:
        case HME_RFT_AAC:
        case HME_RFT_G711A:
        case HME_RFT_G711U:
            // TODO 拷贝音频数据，加入音频帧队列，等待解码
            break;
    }
}
```

PLAYBACK_RES

字段	类型	含义
video_format	uint32	视频格式
video_fps	uint32	视频帧率
video_width	uint32	视频图像宽
video_height	uint32	视频图像高
audio_format	int32	音频格式
audio_channel	uint32	声道数
audio_sample	uint32	音频采样率
continuous_time	uint64	播放时长

GET_FILE_PARAM

字段	类型	含义
channel	uint32	通道号
file_name	char[]	文件名
offset	int32	播放位置偏移
cb_data	cb_pu_download_file	下载数据回调
data	user_data	用户回调数据

GET_FILE_RES

字段	类型	含义
file_name	char[]	文件名
file_size	uint32	文件长度
md5	char[]	

FIND_PICTURE_PARAM

字段	类型	含义
channel	uint32	通道号
cap_type	PIC_CAP_TYPE	图片抓取类型

search_mode	SEARCH_MODE	查询模式
search_month	char[]	查询月份
start_time	char[]	查询开始时间
end_time	char[]	查询结束时间

FIND_PICTURE_DATA

字段	类型	含义
file_name	char[]	文件名
file_size	int64	图片大小
cap_type	PIC_CAP_TYPE	图片抓取类型
cap_time	char[]	抓取时间

DELETE_PICTURE_PARAM

字段	类型	含义
channel	uint32	通道号
file_name	char[]	文件名

GET_PICTURE_PARAM

字段	类型	含义
channel	uint32	通道号
file_name	char[]	文件名
download_mode	PIC_DOWNLOAD_MODE	
start_time	uint64	
end_time	uint64	
key	char[]	
cb_data	cb_pu_download_picture	
data	user_data	

GET_PICTURE_RES

字段	类型	含义
file_count	uint32	

DETECT_UPNP

字段	类型	含义
result	uint8	返回结果（1 表示成功，0 表示失败）
port	uint16	UPNP 端口
on_line	uint8	是否在线（1 表示在线，0 表示不在线）

NETWORK_INFO

字段	类型	含义
dhcp	int32	DHCP 信息
network_card	char[]	网卡型号
link_state	char[]	连接状态
ip	char[]	当前 IP
mask	char[]	当前子网掩码
gate_way	char[]	当前网关

PT_SYSTEM_INFO

字段	类型	含义
sn	char[]	设备 SN 号
soft_version	char[]	软件版本
hard_version	char[]	硬件版本
net_info	NETWORK_INFO	网络信息（见 NETWORK_INFO）
dns_addr	char[][]	DNS 信息
mode_3g	int32	3G 模式（0 代表禁止，1 代表总是在线，2 代表电话或短信触发）
status_3g	int32	3G 在线状态（1 代表在线，其他代表不在

		线)
device_on3g	int32	3G 是否开启
online_time3g	int32	在线时长
online_count3g	int32	3G 数据流量
service_3g	char	3G 运营商 (CT 代表电信, CU 代表联通, CM 代表移动)
vendor_3g	char	3G 卡类型
upnp	int32	UPNP 模式 (1 代表动态, 0 代表静态)
ext_port	int32	UPNP 端口
plt_url	char[]	平台地址
plt_port	uint16	平台端口
log_in_ret	int32	签到状态
eddns	int32	动态域名, 1=开启, 0=关闭
ddns_status	int32	DDNS 状态 (TRUE 代表已登录, FALSE 代表未登录)
sd_status	int32	SD 卡状态, 1=正常, 0=未工作
sd_size	uint32	SD 卡容量
sd_free_size	uint32	SD 卡剩余容量
sd_percent	real64	SD 卡空闲百分比
time	char[]	系统时间
hd_date	char[]	硬件时间
runtime	int32	开机时间
free_percent	real64	设备存储器空闲百分比
check_key	int32	有无加密芯片
region	int32	授权地域
spt_platform	int32	是否支持平台, 0=不支持, 1=支持
ver_limit	int32	需求版本 (固件版本号)

ONLINE_USER_DATA

字段	类型	含义
user_name	char[]	用户名
ip	char[]	用户 IP
log_time	char[]	登录时间
online_time	uint32	在线时长

CONFIG_VIDEO

字段	类型	含义
channel	int32	通道号
fps	int32	视频帧率
bitrate	int32	码流大小（范围 16-4096）
stream_type	CODE_STREAM	码流类型
resolution	char[]	分辨率(可选分辨率有 160 * 120、176 * 144、176 * 120、320 * 240、352 * 288、352 * 240、640 * 480、704 * 576、704 * 480、1280 * 720、1280 * 960)

CONFIG_3G

字段	类型	含义
mode_3g	int32	3G 拨号模式
service_3g	int32	3G 运营商
offline_mode	int32	断线模式
time_out	int32	断线时长
limit_tel_mode	bool	电话号码限制
limittel	char[]	被限制的电话号码
sm_on_cmd	char[]	上线指令
sm_off_cmd	char[]	下线指令

CONFIG_NET

字段	类型	含义
dhcp_mode	int32	配置方式
ip	char[]	IP 地址
mask	char[]	子网掩码
gateway	char[]	网关
dns1	char[]	主 DNS
dns2	char[]	备用 DNS
device_name	char[]	设备名称
is_open_wireless	bool	是否启用无线网络
wireless_dhcp_mode	int32	配置方式
wireless_ip	char[]	无线 IP
wireless_mask	char[]	无线子网掩码
wireless_gateway	char[]	无线网关
service_port	uint16	服务器端口
web_port	uint16	web 端口
is_open_nat	bool	是否开启 NAT 穿透

CONFIG_DDNS

字段	类型	含义
enable	bool	是否开启 DDNS
service	char[]	服务商
url	char[]	域名
user_name	char[]	用户名
password	char[]	密码

CONFIG_PPPOE

字段	类型	含义
is_open_pppoe	bool	是否开启 PPP0E 拨号

user_name	char[]	账号
password	char[]	密码
mode	char[]	拨号方式

CONFIG_UPNP

字段	类型	含义
upnp_mode	int32	UPNP 映射方式（自动映射或手动映射）
port	int32	映射端口
ip	char[]	映射 IP（一般无需设置，当有多个外网 IP 时设置）

CONFIG_WIFI

字段	类型	含义
network_type	int32	工作模式
safe_type	int32	安全类型
encode_type	int32	编码类型
password_type	int32	密码类型
wap_name	char[]	网络名称
password	char[]	网络密码

CONFIG_VIDEO_ENCODE

字段	类型	含义
channel	int32	通道号
enable_main	bool	开启主码流
enable_smooth_m	bool	开启平滑变动
fix_bit_m	int32	码率设置（0 代表固定质量，1 代表固定码率）
quality_m	int32	质量设置
qmax_m	int32	最高质量（0-51）

qmin_m	int32	最低质量（0-51）
bit_rate_m	int32	码流大小
width_m	int32	视频宽
height_m	int32	视频高
framerate_m	int32	视频帧率（1-25）
frame_interval_m	int32	帧间隔（1-120）
enable_sec	bool	开启次码流
enable_smooth_s	bool	开启平滑变动
fix_bit_s	int32	码率设置（0 代表固定质量，1 代表固定码率）
quality_s	int32	质量设置
qmax_s	int32	最高质量（0-51）
qmin_s	int32	最低质量（0-51）
bit_rate_s	int32	码流大小
width_s	int32	视频宽
height_s	int32	视频高
framerate_s	int32	视频帧率（1-25）
frame_interval_s	int32	帧间隔（1-120）

CONFIG_AUDIO_ENCODE

字段		含义
channel	int32	通道号
enable	bool	是否开启音频
enable_vbr	bool	是否开启可变码率
enable_denoise	bool	是否开启噪音消除
volume	real64	音量
audio_channel	int32	声道
sample	int32	音频采样率
format	int32	音频编解码格式

quality	int32	编码质量（0-10）
audio_input	int32	音频输入源
bit_mode	int32	

OSD_STRING

字段	类型	含义
enable	bool	开启画面左上角名称显示
x_pos	int32	显示区域左上角 x 坐标
y_pos	int32	显示区域左上角 y 坐标
osd_str	char[]	显示字符串

OSD_BITRATE

字段	类型	含义
enable	bool	开启画面左上角码率显示
x_pos	int32	显示区域左上角 x 坐标
y_pos	int32	显示区域左上角 y 坐标

OSD_TIME

字段	类型	含义
enable	bool	开启画面左上角时间显示
time_format	int32	时间格式（0 代表的格式为 YY-MM-DD HH:MM:SS、1 代表的格式为 HH:MM:SS YY-MM-DD）
x_pos	int32	显示区域左上角 x 坐标
y_pos	int32	显示区域左上角 y 坐标

CONFIG_OSD

字段	类型	含义
----	----	----

channel	int32	通道号
front_color	int32	前景色
back_color	int32	背景色
transp	int32	透明度
osd_string1	OSD_STRING	显示名称 1
osd_string2	OSD_STRING	显示名称 2
osd_bitrate	OSD_BITRATE	显示帧率
osd_time	OSD_TIME	显示时间

CONFIG_IMAGE_AND_PTZ

字段	类型	含义
channel	int32	通道号
brightness	int32	亮度
contrast	int32	对比度
saturation	int32	饱和度
flip	int32	图像翻转
effects	int32	图像特效
scene	int32	场景
enable_night	bool	是否启用夜间模式
ptz_protocol	char[]	云台协议
baud_rate	int32	云台串口波特率
addr	int32	云台位置
reverse	int32	图像翻转
inout	int32	
leftright	int32	
updown	int32	

MOTION_AREA

字段	类型	含义
enable	bool	是否开启移动侦测
left_top_x	int32	左上角 x 坐标
left_top_y	int32	左上角 y 坐标
right_bottom_x	int32	右下角 x 坐标
right_bottom_y	int32	右下角 y 坐标
threshold	int32	阈值
sensitivity	int32	灵敏度

CONFIG_MONTION_AREA

字段	类型	含义
channel	int32	通道号
frame_interval	int32	侦测时间间隔
sensitivity	int32	灵敏度
area1	MOTION_AREA	侦测区域 1
area2	MOTION_AREA	侦测区域 2
area3	MOTION_AREA	侦测区域 3

ARMING_TIME

字段	类型	含义
start_time	char[]	开始布防时间
end_time	char[]	结束布防时间

ARMING_WEEK_TIME

字段	类型	含义
week_day	int32	周几布防（多个选择时采用或运算）
arming_time	ARMING_TIME []	布防时间段

CONFIG_MOTION_TIME

字段	类型	含义
channel	int32	通道号
enable	bool	是否开启移动侦测
time	ARMING_WEEK_TIME	侦测时间

CONFIG_ALARM_POLICY

字段	类型	含义
enable_video_record	bool	是否开启联动录像
record_time_long	uint32	录像时长
enable_capture_image	bool	是否开启联动抓图
capture_count	uint32	抓拍次数
capture_interval_time	uint32	抓拍时间间隔
enable_gpio_out	uint32	报警端口输出
gpio_time_long	uint32	输出时长
elect_level	uint32	输出电频（常开、常闭）
enable_ftp	bool	是否开启 FTP 上传
ftp_start_time	uint32	开始时间
ftp_end_time	uint32	结束时间
ftp_port	uint32	端口号
ftp_connect_mode	uint32	连接模式（主动、被动）
ftp_user_name	char[]	用户名
ftp_password	char[]	密码
ftp_ip	char[]	FTP 地址
ftp_dir	char[]	FTP 上传目录
enable_email	bool	是否开启邮件功能
enable_send_cap	bool	是否开启抓图功能
enable_email_auth	bool	是否开启服务器密码认证
email_port	uint16	邮件服务器端口

email_interval	uint32	发送间隔时间
email_from	char[]	发件人
email_to	char[]	收件人
email_user	char[]	发件人用户名
email_password	char[]	发件人密码
email_server	char[]	邮件服务器地址

CONFIG_TIME_RECORD

字段	类型	含义
channel	int32	通道号
enable	bool	是否开启定时录像
audio_record	bool	是否录制音频
check	int32[][]	守时录像时间段

CONFIG_GPIO_TIME

字段	类型	含义
channel	int32	通道号
enable	bool	是否开启 GPIO 时间
elect_level	int32	电平（常开或常闭）
time	ARMING_WEEK_TIME	开启时间段

CONFIG_USER_MANAGER

字段	类型	含义
enable	bool	是否开启用户验证
user_info	USER_MANAGER_INFO []	用户信息

CONFIG_TIME

字段	类型	含义
----	----	----

time_mode	int32	时间设置方式（1 代表手动，2 代表 NTP）
ntp_server	char[]	NTP 服务器
cur_time	char[]	当前时间
time_zone	char[]	时区

STORAGE_INFO

字段	类型	含义
total	uint64	总容量
free	uint64	剩余容量

PREPOINT_INFO

字段	类型	含义
prepoint	char[][]	预置位信息
cruise	uint32[][]	自动巡航设置点
cruise_interval	uint32[]	巡航时间间隔
curise_status	bool	巡航状态

USER_INFO

字段	类型	含义
id	int32	用户 ID
name	cpchar	用户名
nick_name	cpchar	用户昵称
tel	cpchar	电话号码
mobile	cpchar	移动电话号码
address	cpchar	用户地址
reg_date	cpchar	注册时间
last_login_date	cpchar	最后一次登录时间
email	cpchar	邮箱地址

email_valid	bool	邮箱是否可用
actor	int32	用户的设备角色
roles1	cpchar	角色 1
roles2	cpchar	角色 2
user_alarm_service	bool	是否使用报警服务
user_transfer_service	int32	是否使用中转穿透服务（0 代表不使用，其余代表使用）

CHANNEL_INFO

字段	类型	含义
index	int32	通道号
name	cpchar	通道名称

BITMAP_INFO_HEADER

字段	类型	含义
biSize	uint32	文件总大小
biWidth	int32	图像宽
biHeight	int32	图像高
biplanes	uint16	图像颜色平面
biBitCount	uint16	图像位数
biCompression	uint32	是否压缩
biSizeImage	uint32	图像数据大小
biXPelsPerMeter	uint32	水平分辨率
biYPelsPerMeter	uint32	垂直分辨率
biClrUsed	uint32	使用的颜色数
biClrImportant	uint32	重要颜色

BITMAP_INFO_HEADER_EX

字段	类型	含义
bih	BITMAP_INFO_HEADER	通用文件头信息
mask	uint32[]	像素颜色掩码（用于 RGB16_565 格式图像）

YUV_PICTURE

字段	类型	含义
width	uint32	图像宽
height	uint32	图像高
ystripe	uint32	y 通道跨度
ustripe	uint32	u 通道跨度
vstripe	uint32	v 通道跨度
ydata	uint8*	y 通道数据
udata	uint8*	u 通道数据
vdata	uint8*	v 通道数据

RAW_FRAME

字段	类型	含义
type	RAW_FRAME_TYPE	数据帧类型
time_stamp	uint32	时间戳
len	uint32	数据帧长度
buf	uint8*	帧数据
hold	bool	是否对数据有拥有权（有拥有权的 Raw_Frame 需自行释放缓冲区）

LOCAL_RECORD_PARAM

字段	类型	含义
file_name	char[]	文件名

ver	uint32	录像主版本号（回放使用）
subver	uint32	录像次版本号（回放使用）
time_cost	uint64	录像时长（回放使用）
video_fmt	VIDEO_ENCODE	视频格式
video_fps	uint32	视频帧率
video_width	uint32	视频宽
video_height	uint32	视频高
audio_fmt	AUDIO_ENCODE	音频格式
audio_channel	uint32	音频通道数
audio_sample	uint32	音频采样率
device_sn	char[]	设备 SN 号
device_name	char[]	设备名称
record_type	VIDEO_RECORD	录像类型

LAN_DEVICE_SEARCH_RES

字段	类型	含义
port	uint16	端口号
ip_addr	char[]	IP 地址
channel_count	uint32	通道数
ip_status	int32	IP 状态
update_status	uint32	更新状态
device_name	char[]	设备名称
device_type	char[]	设备类型
device_sn	char[]	设备 SN 号
mac_addr	char[]	设备 MAC 地址
wifi_mac_addr	char[]	WIFI MAC 地址
lot	char[]	批次
version	char[]	版本号

murl_addr	char[]	多播地址
upnp_status	char[]	UPNP 状态
channel_name	char[]	通道名称
device_key	char[]	设备 KEY
ip_real_addr	char[]	IP 地址

LAN_DEVICE_SEARCH_PARAM

字段	类型	含义
search_mode	LAN_SEARCH_MODE	搜索模式（广播或多播）
host_ip	cpchar	
host_mask	cpchar	
host_gateway	cpchar	
auto_search_interval	uint32	
cb_search	cb_util_lan_device_search	
data	user_data	

cb_util_lan_device_search 回调签名：

```
void LanSearchCallback(user_data data, LAN\_DEVICE\_SEARCH\_RES info,
hm_result result)
{
    // TODO
}
```

枚举值描述

VIDEO_ENCODE

字段	值	含义
HME_VE_H264	1	H264 编码类型
HME_VE_MPEG4	2	MPEG4 编码类型
HME_VE_H265	4	H265 编码类型

AUDIO_ENCODE

字段	值	含义
HME_AE_NONE	-1	
HME_AE_PCMS8	0	PCM 编码类型
HME_AE_ARM	1	ARM 编码类型（保留未用）
HME_AE_SPEEX	2	SPEEX 编码类型
HME_AE_ADPCM	3	ADPCM 编码类型
HME_AE_AAC	4	AAC 编码类型
HME_AE_G711A	5	G711A 编码类型
HME_AE_G711U	6	G711U 编码类型

ALARM_DEVICE_TYPE

字段	值	含义
HME_ADT_VIDEO_DEV	0x01	视频设备
HME_ADT_GPIO_DEV	0x02	GPIO 设备
HME_ADT_STORAGE_DEV	0x04	存储设备
HME_ADT_REMOTECTRL_DEV	0x08	遥控设备
HME_ADT_MAGNETOMETER_DEV	0x10	门磁设备
HME_ADT_INFRARED_DEV	0x20	红外设备
HME_ADT_SMOKE_DEV	0x40	烟感设备
HME_ADT_COAL_DEV	0x80	煤感设备
HME_ADT_WATER_DEV	0x100	水浸设备
HME_ADT_QUAKE_DEV	0x200	震感设备
HME_ADT_PC_CLIENT	0x400	PC 客户端
HME_ADT_PHONE_CLIENT	0x800	手机客户端
HME_ADT_WEB_PLUGIN	0x1000	网页插件
HME_ADT_KEEP_NIGHT_WATCH	0x2000	巡更
HME_ADT_GUARDBAIL	0x4000	护栏

HME_ADT_BOUNDARY	0x8000	周界
HME_ADT_DANGER_BUTTON	0x10000	紧急按钮
HME_ADT_PALTFORM_SERVER	0x20000	平台服务器

ALARM_TYPE

字段	值	含义
HME_AT_SENSOR	0x01	传感器常规报警
HME_AT_MOTION	0x02	移动侦测报警
HME_AT_VIDEO_OCCUSION	0x04	视频遮挡报警
HME_AT_VIDEO_LOSS	0x08	视频丢失报警
HME_AT_SD_FULL	0x10	SD 卡空间满
HME_AT_SD_ERROR	0x20	SD 卡错误报警
HME_AT_SD_NOEXIST	0x40	SD 卡不存在报警
HME_AT_ARM	0x80	布防
HME_AT_DISARM	0x100	撤防
HME_AT_EMERGENCY	0x200	紧急报警
HME_AT_POWER_LOW	0x400	低电压报警
HME_AT_SENSOR_LOSS_HEART	0x800	传感器心跳丢失报警
HME_AT_SENSOR_REMOVED	0x1000	传感器拆除报警
HME_AT_POWER_RESUME	0x2000	电压恢复

CODE_STREAM

字段	值	含义
HME_CS_NONE	-1	无效值
HME_CS_ULTRA_HD	0	超高清（仅 support_third_stream=1 可用）
HME_CS_MAJOR	1	主码流（高画质，高流量）-高清
HME_CS_MINOR	2	次码流（普通画质，低流量）-标清

VIDEO_STREAM

字段	值	含义
HME_VS_NONE	0	无效值
HME_VS_REAL	1	实时视频
HME_VS_PRERECORD	2	录像回放

PTZ_COMMAND

字段	值	含义
HME_PC_NONE	0	无效值
HME_PC_UP	1	上转
HME_PC_LEFT	2	左转
HME_PC_DOWN	3	下转
HME_PC_RIGHT	4	右转
HME_PC_FOCUSUP	7	调焦（增）
HME_PC_FOCUSDOWN	8	调焦（减）
HME_PC_ZOOMIN	9	放大
HME_PC_ZOOMOUT	10	缩小
HME_PC_APTSMALL	11	光圈（减小）
HME_PC_APTBIG	12	光圈（增大）
HME_PC_LIGHTON	13	灯光开
HME_PC_LIGHTOFF	14	灯光关

PTZ_INTERVAL

字段	值	含义
HME_PI_NONE	0	无效值
HME_PI_LR	5	左右巡航

HME_PI_END	6	停止巡航
HME_PI_UDLR	19	上下左右巡航
HME_PI_UD	20	上下巡航

NODE_TYPE_INFO

字段	值	含义
HME_NT_NODE	0	无效值
HME_NT_DEVICE	1	设备节点
HME_NT_DVS	2	DVS 节点
HME_NT_GROUP	3	分组节点
HME_NT_CHANNEL	4	通道节点

SEARCH_MODE

字段	值	含义
HME_SM_NONE	0	无效值
HME_SM_BEG_AND_END_TIME	1	时间段查询
HME_SM_MONTH	2	月份查询

VIDEO_RECORD

字段	值	含义
HME_VR_NONE	0	无效值
HME_VR_HANDLER	1	手动录像
HME_VR_ALARMTYPE	2	报警录像
HME_VR_TIMER	4	定时录像

REMOTE_PLAYBACK_MODE

字段	值	含义
HME_PBM_NONE	0	无效值

HME_PBM_NAME_TIME	1	根据文件名和文件播放时间回放
HME_PBM_START_END	2	从开始时间处开始回放到结束时间处为止
HME_PBM_ALARM_KEY	3	根据报警 key 搜索文件回放
HME_PBM_DOWNLOAD	4	下载远程录像不限速

PIC_CAP_TYPE

字段	值	含义
HME_PCT_NONE	0	无效值
HME_PCT_HANDLER	1	手动抓图
HME_PCT_ALARM_TYPE	2	报警抓图
HME_PCT_TIMER	4	定时抓图

PIC_DOWNLOAD_MODE

字段	值	含义
HME_PDM_NONE	0	无效值
HME_PDM_FILENAME	1	
HME_PDM_START_END_TIME	2	
HME_PDM_KEY	3	

RAW_FRAME_TYPE

字段	值	含义
HME_RFT_P	0	H264 视频帧 P 帧
HME_RFT_I	1	H264 视频帧 I 帧
HME_RFT_AUDIO	2	音频帧
HME_RFT_H265_P	0x20	H265 视频帧 P 帧
HME_RFT_H265_I	0x21	H265 视频帧 I 帧
HME_RFT_PCM	5	音频 PCM

HME_RFT_SPEEX	6	音频 SPEEX
HME_RFT_AAC	7	音频 AAC
HME_RFT_ARM	8	音频 ARM
HME_RFT_ADPCM	0xA	音频 ADPCM
HME_RFT_G711A	0xB	音频 G711A
HME_RFT_G711U	0xC	音频 G711U
HME_RFT_INFO	9	信息帧

BITMAP_FORMAT

字段	值	含义
HME_BF_NODE	0	无效值
HME_BF_RGB32	1	RGB32 格式
HME_BF_BGR32	2	BGR32 格式
HME_BF_RGBA32	3	RGBA32 格式
HME_BF_RGB16_565	4	RGB16_565 格式

USER_ROLE

字段	值	含义
HME_UR_MANAGER	0	管理者
HME_UR_OPERATOR	1	操作者
HME_UR_OBSERVER	2	观察者

PB_MODE

字段	值	含义
HME_PM_NONE	0	
HME_PM_ACTIVE	1	主动模式
HME_PM_PASSIVE	2	被动模式

PLAYBACK_RATE

字段	值	含义
HME_PBR_NONE	0	无效值
HME_PBR_1X	1	原始速率（1 倍速）
HME_PBR_2X	2	2 倍速
HME_PBR_4X	4	4 倍速
HME_PBR_8X	8	8 倍速

LAN_SEARCH_MODE

字段	值	含义
HME_LSM_NONE	0	无效值
HME_LSM_BC	1	广播
HME_LSM_MC	2	多播

技术支持&帮助

技术支持邮箱：[sdk@huamaitel.com](mailto: sdk@huamaitel.com)

如对 SDK 有任何疑问或使用问题，欢迎邮件提出。

-END-