

设备(楼宇可视对讲)
网络 SDK 编程指南
V5.1

声 明

非常感谢您购买我公司的产品，如果您有什么疑问或需要请随时联系我们。

- 我们已尽量保证手册内容的完整性与准确性，但也不免出现技术上不准确、与产品功能及操作不相符或印刷错误等情况，如有任何疑问或争议，请以我司最终解释为准。
- 产品和手册将实时进行更新，恕不另行通知。
- 本手册中内容仅为用户提供参考指导作用，请以 SDK 实际内容为准。

目 录

声 明	I
目 录	II
1 SDK 简介	1
2 SDK 版本更新	4
3 函数调用顺序	7
3.1 楼宇可视对讲主要流程	7
3.2 楼宇可视对讲功能接口流程	9
3.2.1 楼宇可视对讲远程参数配置	9
3.2.2 查询门口机注册信息	11
3.3 可视通话信令交互	12
3.4 预览模块	13
3.5 语音对讲模块	14
3.6 报警模块流程	15
3.6.1 报警布防	15
3.6.2 报警监听	16
4 函数调用实例	18
4.1 楼宇可视对讲示例代码	18
5 函数说明	21
5.1 SDK 初始化	21
5.1.1 初始化 SDK <code>NET_DVR_Init</code>	21
5.1.2 释放 SDK 资源 <code>NET_DVR_Cleanup</code>	21
5.1.3 设置网络连接超时时间和连接尝试次数 <code>NET_DVR_SetConnectTime</code>	21
5.1.4 设置重连功能 <code>NET_DVR_SetReconnect</code>	21
5.1.5 设置接收超时时间 <code>NET_DVR_SetRecvTimeOut</code>	22
5.1.6 获取所有 IP，用于支持多网卡接口 <code>NET_DVR_GetLocalIP</code>	22
5.1.7 设置 IP 绑定 <code>NET_DVR_SetValidIP</code>	22
5.1.8 通过解析服务器获取设备的动态 IP 地址和端口号 <code>NET_DVR_GetDVRIPByResolveSvr_EX</code>	22
5.2 实时预览	23
5.2.1 设置播放显示模式 <code>NET_DVR_SetShowMode</code>	23
5.2.2 主码流动态产生一个关键帧 <code>NET_DVR_MakeKeyFrame</code>	24
5.2.3 子码流动态产生一个关键帧 <code>NET_DVR_MakeKeyFrameSub</code>	24
5.2.4 实时预览 <code>NET_DVR_RealPlay_V40</code>	24
5.2.5 停止预览 <code>NET_DVR_StopRealPlay</code>	25
5.2.6 获取预览时用来解码和显示的播放库句柄 <code>NET_DVR_GetRealPlayerIndex</code>	26
5.3 预览显示视频参数配置	26
5.3.1 获取预览视频显示参数 <code>NET_DVR_ClientGetVideoEffect</code>	26
5.3.2 获取预览视频显示参数 <code>NET_DVR_GetVideoEffect</code>	26
5.3.3 设置预览视频显示参数 <code>NET_DVR_ClientSetVideoEffect</code>	27
5.3.4 设置预览视频显示参数 <code>NET_DVR_SetVideoEffect</code>	27
5.4 预览画面叠加字符和图像	27
5.4.1 预览画面叠加字符和图像，Linux 下无此接口 <code>NET_DVR_RigisterDrawFun</code>	27

5.5	预览时播放声音控制	28
5.5.1	设置声音播放模式 <code>NET_DVR_SetAudioMode</code>	28
5.5.2	独占声卡模式下开启声音 <code>NET_DVR_OpenSound</code>	28
5.5.3	独占声卡模式下开启声音 <code>NET_DVR_CloseSound</code>	28
5.5.4	共享声卡模式下开启声音 <code>NET_DVR_OpenSoundShare</code>	29
5.5.5	共享声卡模式下关闭声音 <code>NET_DVR_CloseSoundShare</code>	29
5.5.6	调节播放音量 <code>NET_DVR_Volume</code>	29
5.6	实时预览数据捕获	29
5.6.1	注册回调函数, 捕获实时码流数据 <code>NET_DVR_SetRealDataCallBack</code>	29
5.6.2	捕获数据并保存到指定的文件中 <code>NET_DVR_SaveRealData</code>	30
5.6.3	停止数据捕获 <code>NET_DVR_StopSaveRealData</code>	30
5.7	预览时抓图	31
5.7.1	设置抓图模式 <code>NET_DVR_SetCapturePictureMode</code>	31
5.7.2	预览时, 单帧数据捕获图片并保存成文件 <code>NET_DVR_CapturePicture</code>	31
5.8	语音对讲	31
	语音对讲(Windows 32 位系统支持)	31
5.8.1	启动语音对讲 <code>NET_DVR_StartVoiceCom_V30</code>	31
5.8.2	设置语音对讲客户端的音量 <code>NET_DVR_SetVoiceComClientVolume</code>	33
5.8.3	停止语音对讲或者语音转发 <code>NET_DVR_StopVoiceCom</code>	33
	音频压缩参数	33
5.8.4	获取当前生效的对讲音频压缩参数 <code>NET_DVR_GetCurrentAudioCompress</code>	33
5.8.5	获取通道参数 <code>NET_DVR_GetDVRConfig</code>	33
5.9	异常消息回调	34
5.9.1	注册接收异常、重连等消息回调函数 <code>NET_DVR_SetExceptionCallBack_V30</code>	34
5.10	SDK 日志和信息	36
5.10.1	获取 SDK 的版本号和 build 信息 <code>NET_DVR_GetSDKBuildVersion</code>	36
5.10.2	获取当前 SDK 的状态信息 <code>NET_DVR_GetSDKState</code>	36
5.10.3	获取当前 SDK 的功能信息 <code>NET_DVR_GetSDKAbility</code>	36
5.10.4	获取 SDK 本地参数 <code>NET_DVR_GetSDKLocalCfg</code>	37
5.10.5	设置 SDK 本地参数 <code>NET_DVR_SetSDKLocalCfg</code>	37
5.10.6	启用写日志文件 <code>NET_DVR_SetLogToFile</code>	38
5.11	获取错误信息	38
5.11.1	返回最后操作的错误码 <code>NET_DVR_GetLastError</code>	38
5.11.2	返回最后操作的错误码信息 <code>NET_DVR_GetErrorMsg</code>	38
5.12	用户注册	39
5.12.1	用户注册设备 <code>NET_DVR_Login_V30</code>	39
5.12.2	用户注销 <code>NET_DVR_Logout</code>	39
5.13	设备能力集	39
5.13.1	获取设备能力集 <code>NET_DVR_GetDeviceAbility</code>	39
5.14	布防、撤防	41
	设置报警等信息上传的回调函数	41
5.14.1	注册回调函数, 接收设备报警消息 <code>NET_DVR_SetDVRMessageCallBack_V30</code>	41
	布防撤防	42
5.14.2	建立报警上传通道, 获取报警等信息 <code>NET_DVR_SetupAlarmChan_V41</code>	42

5.14.3	撤销报警上传通道 NET_DVR_CloseAlarmChan_V30.....	42
5.15	监听报警.....	42
5.15.1	启动监听, 接收设备主动上传的报警等信息 NET_DVR_StartListen_V30	42
5.15.2	停止监听 (支持多线程) NET_DVR_StopListen_V30.....	43
5.16	通用参数配置.....	44
5.16.1	获取设备的配置信息 NET_DVR_GetDVRConfig	44
5.16.2	设置设备的配置信息 NET_DVR_SetDVRConfig	44
5.17	可视对讲参数配置.....	45
5.17.1	获取设备的配置信息 NET_DVR_GetDVRConfig	45
5.17.2	设置设备的配置信息 NET_DVR_SetDVRConfig	46
5.18	批量参数配置.....	47
5.18.1	批量参数获取 (获取 FTP 信息、获取流来源信息) NET_DVR_GetDeviceConfig	47
5.18.2	批量设置参数 (设置 FTP 信息、设置流来源信息) NET_DVR_SetDeviceConfig	48
5.19	查询门口机注册信息.....	50
5.19.1	启动远程配置 (获取门口机注册信息、可视话对讲信令处理) NET_DVR_StartRemoteConfig	50
5.19.2	逐个获取查找到的信息 NET_DVR_GetNextRemoteConfig	51
5.19.3	发送长连接数据 (可视话对讲信令处理) NET_DVR_SendRemoteConfig	52
5.19.4	关闭长连接 NET_DVR_StopRemoteConfig	52
5.20	可视对讲远程控制.....	53
	远程控制.....	53
5.20.1	远程控制 NET_DVR_RemoteControl	53
5.21	报警门禁相关参数配置.....	53
5.21.1	获取报警门禁相关参数 NET_DVR_GetDVRConfig	53
5.21.2	设置报警门禁相关参数 NET_DVR_SetDVRConfig	54
5.21.3	启动远程配置 (获取设置卡参数) NET_DVR_StartRemoteConfig	54
5.21.4	发送长连接数据 (获取设置卡参数) NET_DVR_SendRemoteConfig	56
5.21.5	关闭长连接 NET_DVR_StopRemoteConfig	56
5.22	透明通道.....	57
5.22.1	建立透明通道 NET_DVR_SerialStart_V40	57
5.22.2	通过透明通道向设备串口发送数据 NET_DVR_SerialSend.....	57
5.22.3	断开透明通道 NET_DVR_SerialStop	58
5.23	设备维护.....	58
	远程升级.....	58
5.23.1	设置远程升级时网络环境 NET_DVR_SetNetworkEnvironment	58
5.23.2	远程升级 NET_DVR_Upgrade	58
5.23.3	获取远程升级的进度 NET_DVR_GetUpgradeProgress.....	59
5.23.4	获取远程升级的状态 NET_DVR_GetUpgradeState	59
5.23.5	获取远程升级的阶段信息 NET_DVR_GetUpgradeStep.....	59
5.23.6	关闭远程升级句柄, 释放资源 NET_DVR_CloseUpgradeHandle.....	60
	日志查找.....	60
5.23.7	查找设备的日志信息 NET_DVR_FindDVRLog_V30.....	60
5.23.8	逐条获取查找到的日志信息 NET_DVR_FindNextLog_V30	60
5.23.9	释放查找日志的资源 NET_DVR_FindLogClose_V30	61
	远程恢复默认参数、重启	61

5.23.10	恢复设备默认参数 <i>NET_DVR_RestoreConfig</i>	61
5.23.11	重启设备 <i>NET_DVR_RebootDVR</i>	61
	配置文件导入导出.....	62
5.23.12	导出配置文件 <i>NET_DVR_GetConfigFile</i>	62
5.23.13	导入配置文件 <i>NET_DVR_SetConfigFile</i>	62
6	错误代码及说明	63
6.1	网络通讯库错误码.....	63
6.2	RTSP 通讯库错误码.....	67
6.3	软解码库错误码.....	67
7	结构体说明	69
7.1	宏定义.....	69
7.2	<i>NET_DVR_ACS_PARAM_TYPE</i> : 门禁主机参数结构体	70
7.3	<i>NET_DVR_AGAIN_RELATEDDEV</i> : 二次确认机关联网络设备参数结构体	71
7.4	<i>NET_DVR_ALARMER</i> :报警设备信息	72
7.5	<i>NET_DVR_ALARM_RS485CFG</i> :报警主机 RS485 参数结构体	73
7.6	<i>NET_DVR_ALARMIN_PARAM</i> :防区参数	74
7.7	<i>NET_DVR_CALL_ROOM_CFG</i> :按键呼叫住户配置结构体.....	77
7.8	<i>NET_DVR_CARD_CFG</i> : 卡参数配置结构体	78
7.9	<i>NET_DVR_CARD_CFG_COND</i> :卡参数配置条件结构体	80
7.10	<i>NET_DVR_CARD_CFG_SEND_DATA</i> : 获取卡参数的发送数据.....	80
7.11	<i>NET_DVR_CLIENTINFO</i> :预览参数结构体	81
7.12	<i>NET_DVR_COMPRESSIONCFG_V30</i> :通道压缩参数结构体.....	81
7.13	<i>NET_DVR_COMPRESSION_INFO_V30</i> :码流压缩参数	82
7.14	<i>NET_DVR_CONTROL_GATEWAY</i> : 远程开锁控制参数结构体	83
7.15	<i>NET_DVR_DAYTIME</i> :时间参数结构体.....	84
7.16	<i>NET_DVR_DDNS_ADDRESS</i> :设备 DDNS 域名信息	84
7.17	<i>NET_DVR_DDNSPARA</i> : 网络应用参数 (DDNS) 结构体.....	85
7.18	<i>NET_DVR_DELAY_TIME</i> :延时开机参数结构体	86
7.19	<i>NET_DVR_DEVICECFG_V40</i> :设备参数	86
7.20	<i>NET_DVR_DEVICEINFO_V30</i> :设备参数结构体.....	89
7.21	<i>NET_DVR_DIRECT_CONNECT_CHAN_INFO</i> :IP 通道信息.....	92
7.22	<i>NET_DVR_DOOR_CFG</i> 门参数配置结构体	93
7.23	<i>NET_DVR_ELEVATORCONTROL_CFG_V40</i> :可视对讲梯控器参数配置结构体	94
7.24	<i>NET_DVR_ETHERNET_V30</i> :以太网配置参数	95
7.25	<i>NET_DVR_EXCEPTION_V30</i> :异常参数	96
7.26	<i>NET_DVR_EXCEPTION_V40</i> :异常参数配置 (扩展)	97
7.27	<i>NET_DVR_FTPCFG_V40</i> : FTP 配置结构体	97
7.28	<i>NET_DVR_FTP_TYPE</i> : FTP 类型信息结构体	99
7.29	<i>NET_DVR_HANDLEEXCEPTION</i> :报警和异常处理结构体	100
7.30	<i>NET_DVR_HANDLEEXCEPTION_V30</i> :报警和异常处理.....	100
7.31	<i>NET_DVR_HANDLEEXCEPTION_V41</i> :异常参数信息结构体.....	101
7.32	<i>NET_DVR_HIDEALARM</i> :遮挡报警参数结构体	101
7.33	<i>NET_DVR_HOLIDAY_PARAM_CFG</i> :假日参数配置结构体	102
7.34	<i>NET_DVR_HOLIDAY_PARAM</i> :假日参数结构体	103

7.35	NET_DVR_HOLIDATE_UNION:假日日期结构联合体	103
7.36	NET_DVR_HOLIDATE_MODEA:假日日期模式 A	104
7.37	NET_DVR_HOLIDATE_MODEB:假日日期模式 B.....	104
7.38	NET_DVR_HOLIDATE_MODEC:假日日期模式 C.....	105
7.39	NET_DVR_IP_ADDRESS:设备 IP 地址信息	105
7.40	NET_DVR_IPADDR:IP 地址	106
7.41	NET_DVR_LOCAL_ABILITY_PARSE_CFG:能力集解析库配置	106
7.42	NET_DVR_LOCAL_BYTE_ENCODE_CONVERT:字符编码转换参数结构体.....	107
7.43	NET_DVR_LOCAL_CHECK_DEV:设备在线巡检参数结构体。	108
7.44	NET_DVR_LOCAL_MEM_POOL_CFG:内存池本地配置	108
7.45	NET_DVR_LOCAL_MODULE_RECV_TIMEOUT_CFG:按模块配置超时时间.....	109
7.46	NET_DVR_LOCAL_PROTECT_KEY_CFG:密钥配置	109
7.47	NET_DVR_LOCAL_TALK_MODE_CFG:对讲模式配置.....	110
7.48	NET_DVR_LOCAL_TCP_PORT_BIND_CFG:本地 TCP 端口绑定配置	110
7.49	NET_DVR_LOCAL_UDP_PORT_BIND_CFG:本地 UDP 端口绑定配置	111
7.50	NET_DVR_LOG_V30:日志信息	111
7.51	NET_DVR_MANAGE_UNIT_DEVICEID:管理机编号参数结构体	113
7.52	NET_DVR_MANAGE_UNIT_OPERATION_TIME_CFG: 管理机操作时间配置结构体.....	113
7.53	NET_DVR_MANAGE_UNIT_RELATEDDEV: 管理机关联网络设备参数结构体	114
7.54	NET_DVR_MOTION:移动侦测参数结构体	114
7.55	NET_DVR_NETAPPCFG: 网络应用参数（NTP/DDNS）结构体	115
7.56	NET_DVR_NETCFG_V30:网络配置	115
7.57	NET_DVR_NOTICE_DATA:公告数据下发参数	117
7.58	NET_DVR_NOTICEDATA_RECEIPT_INFO:公告信息阅读回执信息	118
7.59	NET_DVR_NOTICE_PIC:公告图片信息	118
7.60	NET_DVR_NTPPARA:网络应用参数(NTP)	119
7.61	NET_DVR_OUTDOOR_FENCE_DEVICEID:围墙机编号参数结构体	119
7.62	NET_DVR_OUTDOOR_UNIT_DEVICEID: 门口机编号参数结构体.....	120
7.63	NET_DVR_OUTDOOR_UNIT_OPERATION_TIME_CFG: 门口机/围墙机/别墅门口机/二次确认机操作时间配置结构体.....	120
7.64	NET_DVR_OUTDOOR_UNIT_RELATEDDEV: 门口机/围墙机/别墅门口机关联网络设备参数结构体	121
7.65	NET_DVR_PICCFG:通道图像结构体.....	121
7.66	NET_DVR_PICTURE_NAME_EX: 图片命名规则扩展.....	123
7.67	NET_DVR_PPPCFG_V30:PPP 参数结构体	124
7.68	NET_DVR_PPPOECFG:PPPoE 配置结构体	125
7.69	NET_DVR_PPPOECFG: 预览参数结构体.....	126
7.70	NET_DVR_PRIVILEGE_PASSWORD_CFG:权限密码配置	127
7.71	NET_DVR_PU_STREAM_URL:URL 取流配置.....	128
7.72	NET_DVR_PU_STREAM_URL_CFG:URL 取流路径构成参数	129
7.73	NET_DVR_RECTCFG:窗口设置结构体	130
7.74	NET_DVR_RECTCFG_EX:矩形窗口参数	130
7.75	NET_DVR_RS232CFG_V30:RS232 串口参数结构体.....	131
7.76	NET_DVR_SCHEDTIME:起止时间段参数	131
7.77	NET_DVR_SDKSTATE:SDK 状态信息	132

7.78	NET_DVR_SDKABL:SDK 功能信息.....	133
7.79	NET_DVR_SERIAL_CONTROL:串口信息结构体	134
7.80	NET_DVR_SETUPALARM_PARAM:报警布防参数.....	134
7.81	NET_DVR_SINGLE_RS232:RS232 串口参数	135
7.82	NET_DVR_SIREN_PARAM:警号参数结构体	136
7.83	NET_DVR_STREAM_INFO:流信息	136
7.84	NET_DVR_STREAM_MODE_TYPE:取流方式.....	137
7.85	NET_DVR_STREAM_TYPE_UNION:取流方式联合体	137
7.86	NET_DVR_STREAM_SRC_CFG:流的来源配置	138
7.87	NET_DVR_STREAM_SRC_INFO:流的来源信息	138
7.88	NET_DVR_STRUCTHEAD: 头信息结构体	139
7.89	NET_DVR_SWITCH_DAY_TIME: 开关机计划参数结构体.....	139
7.90	NET_DVR_SWITCH_TIME_POINT: 按时间点开关机参数结构体	140
7.91	NET_DVR_SYSTEM_TIME:时间信息结构体	140
7.92	NET_DVR_TIME:时间参数	141
7.93	NET_DVR_TIME_EX:时间参数	141
7.94	NET_DVR_TIMEPOINT:时间点参数	142
7.95	NET_DVR_TIME_SWITCH_CFG:时间点参数	143
7.96	NET_DVR_UNLOCK_RECORD_INFO:可视对讲开锁记录信息.....	143
7.97	NET_DVR_USER_INFO_V40: 用户信息结构体	144
7.98	NET_DVR_USER_V40: 用户参数配置（扩展）结构体	146
7.99	NET_VCA_BEHAVIOR_ABILITY:行为分析能力输出参数结构体	147
7.100	NET_VCA_DEV_ABILITY:智能设备能力集结构体	148
7.101	NET_DVR_VALID_PERIOD_CFG: 有效期参数结构体.....	150
7.102	NET_DVR_VIDEOINTERCOM_STREAM: 可视对讲流通道参数结构体.....	151
7.103	NET_DVR_VIDEO_CALL_COND:可视对讲信令处理条件参数.....	152
7.104	NET_DVR_VIDEO_CALL_PARAM:可视对讲信令处理参数.....	152
7.105	NET_DVR_VIDEO_INTERCOM_ALARM:可视对讲报警信息.....	154
7.106	NET_DVR_VIDEO_INTERCOM_ALARM_INFO_UNION:可视对讲报警信息联合体.....	154
7.107	NET_DVR_VIDEO_INTERCOM_ALARM_CFG: 可视对讲报警事件参数结构体	155
7.108	NET_DVR_VIDEO_INTERCOM_DEVICEID_CFG: 可视对讲设备编号配置结构体	155
7.109	NET_DVR_VIDEO_INTERCOM_EVENT: 可视对讲事件记录信息结构体.....	156
7.110	NET_DVR_VIDEO_INTERCOM_EVENT_INFO_UNION: 可视对讲事件记录信息联合体	157
7.111	NET_DVR_VIDEO_INTERCOM_IOIN_CFG:可视对讲 IO 输入参数配置结构体	157
7.112	NET_DVR_VIDEO_INTERCOM_IOOUT_CFG:可视对讲 IO 输出参数配置结构体.....	158
7.113	NET_DVR_VIDEO_INTERCOM_UNIT_DEVICEID_UNION: 可视对讲设备编号参数联合体	158
7.114	NET_DVR_VIDEO_INTERCOM_OPERATION_TIME_CFG: 操作时间配置结构体.....	159
7.115	NET_DVR_VIDEO_INTERCOM_OPERATION_TIME_UNION: 可视对讲操作时间配置联合体	160
7.116	NET_DVR_VIDEO_INTERCOM_RELATEDDEV_CFG: 可视对讲关联网络设备配置结构体.....	160
7.117	NET_DVR_VIDEO_INTERCOM_UNIT_RELATEDDEV_UNION: 可视对讲关联网络设备配置联合体	161
7.118	NET_DVR_VIHOST:信号丢失报警参数结构体	161
7.119	NET_DVR_VIS_REGISTER_INFO: 门口机注册信息结构体	162
7.120	NET_DVR_VCS_USER_INFO:用户信息结构体.....	163
7.121	NET_DVR_VIDEOEFFECT:视频参数.....	164

7.122	NET_DVR_VOLUME_CFG:音量调节参数配置.....	165
7.123	NET_DVR_WDR:宽动态参数结构体	165
7.124	NET_DVR_WDR_CFG:宽动态参数配置结构体.....	166
7.125	NET_DVR_ZONE_ALARM_INFO:防区报警信息	166
7.126	NET_DVR_ZONEANDDST:夏令时参数	167
7.127	IP 可视对讲主机能力集能力 XML 描述.....	168
7.128	日志类型	178

1 SDK 简介

设备网络 SDK 是基于设备私有网络通信协议开发的，为嵌入式网络硬盘录像机、视频服务器、网络摄像机、网络球机、解码器、多屏控制器、报警主机、楼宇可视对讲等产品服务的配套模块，用于远程访问和控制设备软件的二次开发。本文档主要介绍楼宇可视对讲相关的功能。

设备网络 SDK 包含网络通讯库、软解码库、硬解码库等功能组件，我们提供 Windows 和 Linux 两个版本的 SDK，各自所包含的组件如下：

表 1.1 Windows SDK 组件

网络通讯库	外部接口	HCNetSDK.h	头文件	
		HCNetSDK.lib	LIB 库文件	
		HCNetSDK.dll	DLL 库文件	
	核心组件	HCCore.lib	LIB 库文件	
		HCCore.dll	DLL 库文件	
组件库	设备配置核心组件	HCCoreDevCfg.dll	DLL 库文件	HCNetSDKCom 文件夹
	预览组件	HCPreview.lib	LIB 库文件	HCNetSDKCom 文件夹
		HCPreview.dll	DLL 库文件	HCNetSDKCom 文件夹
	回放组件	HCPlayBack.dll	DLL 库文件	HCNetSDKCom 文件夹
	语音组件	HCVoiceTalk.dll	DLL 库文件	HCNetSDKCom 文件夹
	报警组件	HCAalarm.lib	LIB 库文件	HCNetSDKCom 文件夹
		HCAalarm.dll	DLL 库文件	HCNetSDKCom 文件夹
	显示组件	HCDisplay.dll	DLL 库文件	HCNetSDKCom 文件夹
	行业应用管理配置组件	HCIndustry.dll	DLL 库文件	HCNetSDKCom 文件夹
	维护管理配置组件	HCGeneralCfgMgr.lib	LIB 库文件	HCNetSDKCom 文件夹
		HCGeneralCfgMgr.dll	DLL 库文件	HCNetSDKCom 文件夹
RTSP 通讯库		StreamTransClient.dll	DLL 库文件	HCNetSDKCom 文件夹
转封装库		SystemTransform.dll	DLL 库文件	HCNetSDKCom 文件夹
语音对讲库		AudioIntercom.dll	DLL 库文件	HCNetSDKCom 文件夹
		OpenAL32.dll	DLL 库文件	HCNetSDKCom 文件夹
字符转码库		libiconv2.dll	DLL 库文件	HCNetSDKCom 文件夹
模拟能力集		LocalXml.zip	XML 文件包	
软解码库		plaympeg4.h	头文件	
		PlayCtrl.lib	LIB 库文件	
		PlayCtrl.dll	DLL 库文件	
		AudioRender.dll	DLL 库文件	

		SuperRender.dll	DLL 库文件	
硬解码库		DsSdk.dll	DLL 库文件	HCNetSDKCom 文件夹

表 1.2 Linux SDK 组件

网络通讯库	外部接口	HCNetSDK.h	头文件	
		libhcnetsdk.so	SO 库文件	
	核心组件	libHCCore.so	SO 库文件	
组件库	设备配置核心组件	libHCCoreDevCfg.so	SO 库文件	HCNetSDKCom 文件夹
	预览组件	libHCPreview.so	SO 库文件	HCNetSDKCom 文件夹
	回放组件	libHCPlayBack.so	SO 库文件	HCNetSDKCom 文件夹
	语音组件	libHCVoiceTalk.so	SO 库文件	HCNetSDKCom 文件夹
	报警组件	libHCAalarm.so	SO 库文件	HCNetSDKCom 文件夹
	显示组件	libHCDisplay.so	SO 库文件	HCNetSDKCom 文件夹
	行业应用管理配置组件	libHCIndustry.so	SO 库文件	HCNetSDKCom 文件夹
	维护管理配置组件	libHCGeneralCfgMgr.so	SO 库文件	HCNetSDKCom 文件夹
hpr 库		libhpr.so	SO 库文件	
RTSP 通讯库		libStreamTransClient.so	SO 库文件	HCNetSDKCom 文件夹
转封装库		libSystemTransform.so	SO 库文件	HCNetSDKCom 文件夹
字符转码库		libiconv2.so	SO 库文件	HCNetSDKCom 文件夹
软解码库		LinuxPlayM4.h	头文件	
		PlayM4.h	头文件	
		libMPCtrl.so	SO 库文件	
		libPlayCtrl.so	SO 库文件	

本版本的设备网络 SDK 开发包中包含以上各个组件，**HCNetSDK.dll、HCCore.dll 必须加载**（对于 Linux SDK，即 libhcnetsdk.so、libHCCore.so），其他组件，用户可以根据需要选择其中的一部分或者全部，以下将对各个组件在 SDK 中的作用和使用条件分别说明。

- **网络通讯库**：设备网络 SDK 的主体，主要用于网络客户端与各类产品之间的通讯交互，负责远程功能调控，远程参数配置及码流数据的获取和处理等。设备网络 SDK V5.0 针对产品应用业务进行细化，对之前版本的 SDK 的功能模块进行组件化，其中外部接口（HCNetSDK.dll）仍然保持和设备网络 SDK V4.x 版本保存一致（向下兼容），其他单独的业务功能（预览、回放等）可以加载单独的模块组件，多个业务功能也可以组合使用。**更新 SDK 时，HCNetSDK.dll、HCCore.dll 以及 HCNetSDKCom 文件夹下的功能组件库文件都需要更新加载，且 HCNetSDKCom 文件夹名不能修改。**
- **hpr 库**：网络通讯库的依赖库，Linux SDK 使用时和网络通讯库同时加载。
- **RTSP 通讯库**：支持 RTSP 传输协议的网络库。当需要对支持 RTSP 协议的产品进行取流等操作时就必须加载该项组件。

- **转封装库：**库的功能可以分为两种：一种是将标准码流转换成采用我们公司封装格式的码流。当用户需要对支持 RTSP 协议的产品捕获采用本公司封装格式的码流数据时（即当设置 NET_DVR_RealPlay_V40 接口中的回调函数捕获数据或者调用 NET_DVR_SetRealDataCallBack 接口捕获数据时）必须加载该组件。另一种功能是将标准码流转换成其他格式的封装，如 3GPP、PS 等。例如，当用户需要对支持 RTSP 协议的产品实时捕获指定封装格式的码流数据（对应的 SDK 接口为 NET_DVR_SaveRealData）时必须加载该项组件。
- **语音对讲库：**用于语音对讲时通过声卡采集数据并按照指定的编码格式编码码流或者解码播放音频码流数据（不带封装格式的码流数据）。V4.2.2.5 及以前版本 SDK 均采用 windows API 实现相关功能。之后版本默认使用语音对讲库的方式，通过接口 [NET_DVR_SetSDKLocalCfg](#) 可以选择之前的 windows API 模式。OpenAL32.dll 为依赖库，语音对讲库模式下必须加载。**Windows64 位或者 Linux 系统下无语音对讲功能。CVR 暂不支持语音对讲功能。**
- **字符转换库：**电脑字符集和设备字符集不一致时，SDK 内部需要进行字符编码转换，SDK 默认使用 libiconv 库进行类型转换。如果用户不想使用 libiconv 编码库，可以调用 [NET_DVR_SetSDKLocalCfg](#) (类型: NET_SDK_LOCAL_CFG_TYPE_BYTE_ENCODE) 设置字符转码回调函数，将用户自己的字符编码接口告知 SDK，然后 SDK 将使用用户提供的字符编码接口进行字符串处理。
- **模拟能力集：**如果需要获取设备能力集（[NET_DVR_GetDeviceAbility](#)），建议调用 [NET_DVR_SetSDKLocalCfg](#) 启用模拟能力集，此时需要加载 LocalXml.zip（要求和网络通讯库放在同一个目录下）。
- **软解码库：**主要用于对实时码流数据进行解码显示（实现预览功能）和对录像文件进行回放解码等。用户如果需要在 SDK 内部进行对实时流和录像码流播放显示时（即 NET_DVR_RealPlay_V40 接口的第二个结构体参数的播放句柄设置成有效句柄时）必须加载该组件，而如果用户仅需要用网络通讯库捕获到数据后再外部自行处理就不需要加载该组件，这种情况下用户在外自行解码将更灵活，可参见软解码库函数说明《播放器 SDK 编程指南》。**Linux 64 位系统不支持软解码功能，预览、回放等窗口句柄传空，仅支持只取流不解码。**
- **硬解码库：**需在配备硬解码卡（MD 卡）的前提下使用，通过解码卡的解码与输出功能实现实时流的解码显示及向监视器上矩阵输出的功能。**Windows64 位或者 Linux 系统下无该硬解码库。对于 LCD 屏，不需要使用该库文件。**

2 SDK 版本更新

Version 5.1.3.10(build20150720)

- 门口机：DS-KD8102-2、DS-KD8002-2A
管理机：DS-KM8301
别墅门口机：DS-KV8102-1C、DS-KV8102-2C、DS-KV8102-1A、DS-KV8202-1A、DS-KV8402-1A
- 可视对讲部分
 - ✧ 参数配置
 - NET_DVR_SetDVRConfig(命令：NET_DVR_SET_VIDEO_INTERCOM_DEVICEID_CFG)，通过结构体 [NET_DVR_VIDEO_INTERCOM_DEVICEID_CFG](#) 可以设置可视对讲设备编号信息。可视对讲设备编号参数可设置门口机设备参数编号或者管理机设备编号参数或者围墙机设备编号参数以及别墅门口机设备编号参数；
 - 设置权限密码配置信息，通过 NET_DVR_SetDVRConfig（命令：NET_DVR_SET_PRIVILEGE_PASSWORD），通过结构体 [NET_DVR_PRIVILEGE_PASSWORD_CFG](#) 可以设置工程密码或者布/撤防密码/户主开锁密码/防劫持密码等参数；
 - 设置操作时间配置，可以调用接口 NET_DVR_SetDVRConfig（命令：NET_DVR_SET_OPERATION_TIME_CFG），通过结构体 [NET_DVR_VIDEO_INTERCOM_OPERATION_TIME_CFG](#) 可以配置不同设备类型的操作时间；
 - 设置关联网络设备参数，可以调用接口 NET_DVR_SetDVRConfig（命令：NET_DVR_SET_VIDEO_INTERCOM_RELATEDDEV_CFG），通过结构体 [NET_DVR_VIDEO_INTERCOM_RELATEDDEV_CFG](#) 配置下不同设备关联网络设备配置参数；
 - 设置可视对讲流通道参数，可以调用接口 NET_DVR_SetDVRConfig（命令：NET_DVR_SET_VIDEOINTERCOM_STREAM），通过结构体 [NET_DVR_VIDEOINTERCOM_STREAM](#) 配置可视对讲设备视频源信息；
 - 设置可视对讲 IO 输入参数，可以调用接口 NET_DVR_SetDVRConfig（命令：NET_DVR_SET_VIDEO_INTERCOM_IOIN_CFG），通过结构体 [NET_DVR_VIDEO_INTERCOM_IOIN_CFG](#) 配置可视对讲 IO 输入参数；
 - 设置可视对讲 IO 输出参数，可以调用接口 NET_DVR_SetDVRConfig（命令：NET_DVR_SET_VIDEO_INTERCOM_IOOUT_CFG），通过结构体 [NET_DVR_VIDEO_INTERCOM_IOOUT_CFG](#) 配置可视对讲 IO 输出参数；
 - 设置梯控器参数，可以调用接口 NET_DVR_SetDVRConfig（命令：NET_DVR_SET_ELEVATORCONTROL_CFG_V40），通过结构体 [NET_DVR_ELEVATORCONTROL_CFG_V40](#) 配置梯控参数；
 - 设置宽动态参数，可以调用接口 NET_DVR_SetDVRConfig（命令：NET_DVR_SET_WDR_CFG），通过结构体 [NET_DVR_WDR_CFG](#) 设置宽动态参数；
 - 设置按键呼叫住户配置参数，可以调用接口 NET_DVR_SetDVRConfig（命令：NET_DVR_SET_CALL_ROOM_CFG），通过结构体 [NET_DVR_CALL_ROOM_CFG](#) 实现按键呼叫住户配置参数；
 - 设置音频输入音量调节参数，可以调用接口 NET_DVR_SetDVRConfig（命令：NET_DVR_SET_AUDIOIN_VOLUME_CFG），通过结构体 [NET_DVR_VOLUME_CFG](#) 实现设置音频输入音量调节参数；
 - 设置音频输出音量调节参数，可以调用接口 NET_DVR_SetDVRConfig（命令：

- NET_DVR_SET_AUDIOOUT_VOLUME_CFG)，通过结构体 [NET_DVR_VOLUME_CFG](#) 实现音频输出音量调节参数；
- 设置可视对讲报警事件参数，可以调用接口 NET_DVR_SetDVRConfig（命令：NET_DVR_SET_VIDEO_INTERCOM_ALARM_CFG），通过结构体 [NET_DVR_VIDEO_INTERCOM_ALARM_CFG](#) 设置可视对讲报警事件参数。
- ✧ 可视对讲信令交互
 - 调用接口 NET_DVR_StartRemoteConfig（命令：NET_DVR_VIDEO_CALL_SIGNAL_PROCESS）(别墅门口机 V1.1 新增)，建立信令交互长连接；
 - 调用接口 NET_DVR_SendRemoteConfig，向设备发送信令；
 - 调用接口 NET_DVR_StopRemoteConfig，关闭长连接，释放资源。
 - ✧ 可视对讲远程控制
 - 调用接口 NET_DVR_RemoteControl（命令：NET_DVR_REMOTECONTROL_GATEWAY），实现远程开锁；
 - 调用接口 NET_DVR_RemoteControl（命令：NET_DVR_REMOTECONTROL_NOTICE_DATA），实现公告信息下发；
 - ✧ 查询门口机注册信息
 - 调用接口 NET_DVR_StartRemoteConfig（命令：NET_DVR_GET_VIS_REGISTER_INFO），启用长连接获取门口机注册信息；
 - 调用接口 NET_DVR_GetNextRemoteConfig（命令：NET_DVR_GET_VIS_REGISTER_INFO），逐个获取查询到的结果信息；
 - 调用接口 NET_DVR_StopRemoteConfig，关闭长连接，释放资源。
- 报警门禁
- ✧ 相关参数配置
 - NET_DVR_SetDVRConfig 接口可以实现：设置 RS485 参数（命令：NET_DVR_SET_ALARM_RS485CFG，对应结构体 [NET_DVR_ALARM_RS485CFG](#)）、设置防区参数（命令：NET_DVR_SET_ALARMIN_PARAM，对应结构体： [NET_DVR_ALARMIN_PARAM](#)）、设置门参数（命令：NET_DVR_SET_DOOR_CFG，对应结构体： [NET_DVR_DOOR_CFG](#)）等；
 - ✧ 卡参数配置
 - 调用接口 NET_DVR_StartRemoteConfig（命令：NET_DVR_GET_CARD_CFG、NET_DVR_SET_CARD_CFG），启动卡参数配置长连接；
 - 调用接口 NET_DVR_SendRemoteConfig（命令：NET_DVR_GET_CARD_CFG、NET_DVR_SET_CARD_CFG），发送卡参数配置数据；
 - 调用接口 NET_DVR_StopRemoteConfig，断开长连接。
 - ✧ 清空门禁卡信息
 - 调用接口 NET_DVR_RemoteControl（命令：NET_DVR_CLEAR_ACS_PARAM），清空门禁卡信息。
- 报警功能
- ✧ 布防报警
 - 布防 NET_DVR_SetupAlarmChan_V41（别墅门口机 V1.1 新增支持），其中支持的报警类型支持报警信息类型：COMM_UPLOAD_VIDEO_INTERCOM_EVENT，对应结构体 [NET_DVR_VIDEO_INTERCOM_EVENT](#)；COMM_ALARM_VIDEO_INTERCOM，对应结构体 [NET_DVR_VIDEO_INTERCOM_ALARM](#) 获取报警信息。
 - 调用接口 NET_DVR_CloseAlarmChan_V30 实现撤防。
 - ✧ 监听报警
 - 开启监听 NET_DVR_StartListen_V30 其中支持的报警类型支持报警信息类型：

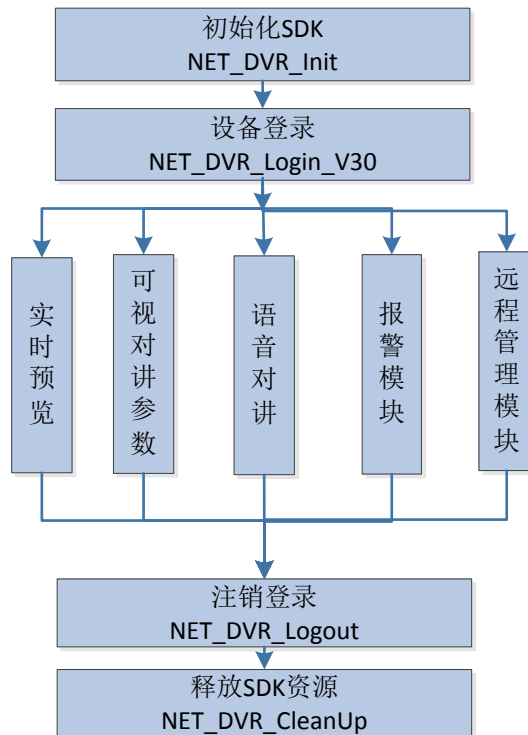
COMM_UPLOAD_VIDEO_INTERCOM_EVENT，对应结构体 [NET_DVR_VIDEO_INTERCOM_EVENT](#)；
COMM_ALARM_VIDEO_INTERCOM，对应结构体 [NET_DVR_VIDEO_INTERCOM_ALARM](#) 获取报警信息。

- 停止监听 NET_DVR_StopListen_V30 实现撤防。

3 函数调用顺序

3.1 楼宇可视对讲主要流程

图 3.1 楼宇可视对讲主要流程



楼宇可视对讲功能模块包括实时预览、报警监听、远程参数配置等功能模块。

- 在初始化 SDK 和登录设备之后，可通过接口 [NET_DVR_GetDeviceAbility](#)（能力集类型：IP_VIEW_DEV_ABILITY）获取可视对讲能力集，能力集类型：ACS_ABILITY 获取门禁主机能力集，包括软硬件能力集、所有编码能力、用户管理参数能力、网络应用能力、流能力、串口能力、日志搜索能力、通道输入能力集等。
- 设置 FTP 信息，通过接口 [NET_DVR_GetDeviceConfig](#)（命令：NET_DVR_SET_FTPCFG_V40），设置上传的 FTP 参数信息。
- 设置流的来源信息，通过接口 [NET_DVR_SetDeviceConfig](#)（命令：NET_DVR_SET_STREAM_SRC_INFO），设置流的来源参数。
- 设置可视对讲设备编号，通过接口 [NET_DVR_SetDVRConfig](#)（命令：NET_DVR_SET_VIDEO_INTERCOM_DEVICEID_CFG），设置设备类型、是否自动注册以及设备编号参数。
- 设置操作时间配置，通过接口 [NET_DVR_SetDVRConfig](#)（命令：NET_DVR_SET_OPERATION_TIME_CFG），设置管理及或门口机/围墙机/别墅门口机/二次确认及操作时间配置参数。
- 设置关联网络设备参数，通过接口 [NET_DVR_SetDVRConfig](#)（命令：NET_DVR_SET_VIDEO_INTERCOM_RELATEDDEV_CFG），设置门口机/围墙机关联网络设备参数配置或者管理机关联网络，别墅门口机关联网络、二次确认机关联网络设备参数配置。

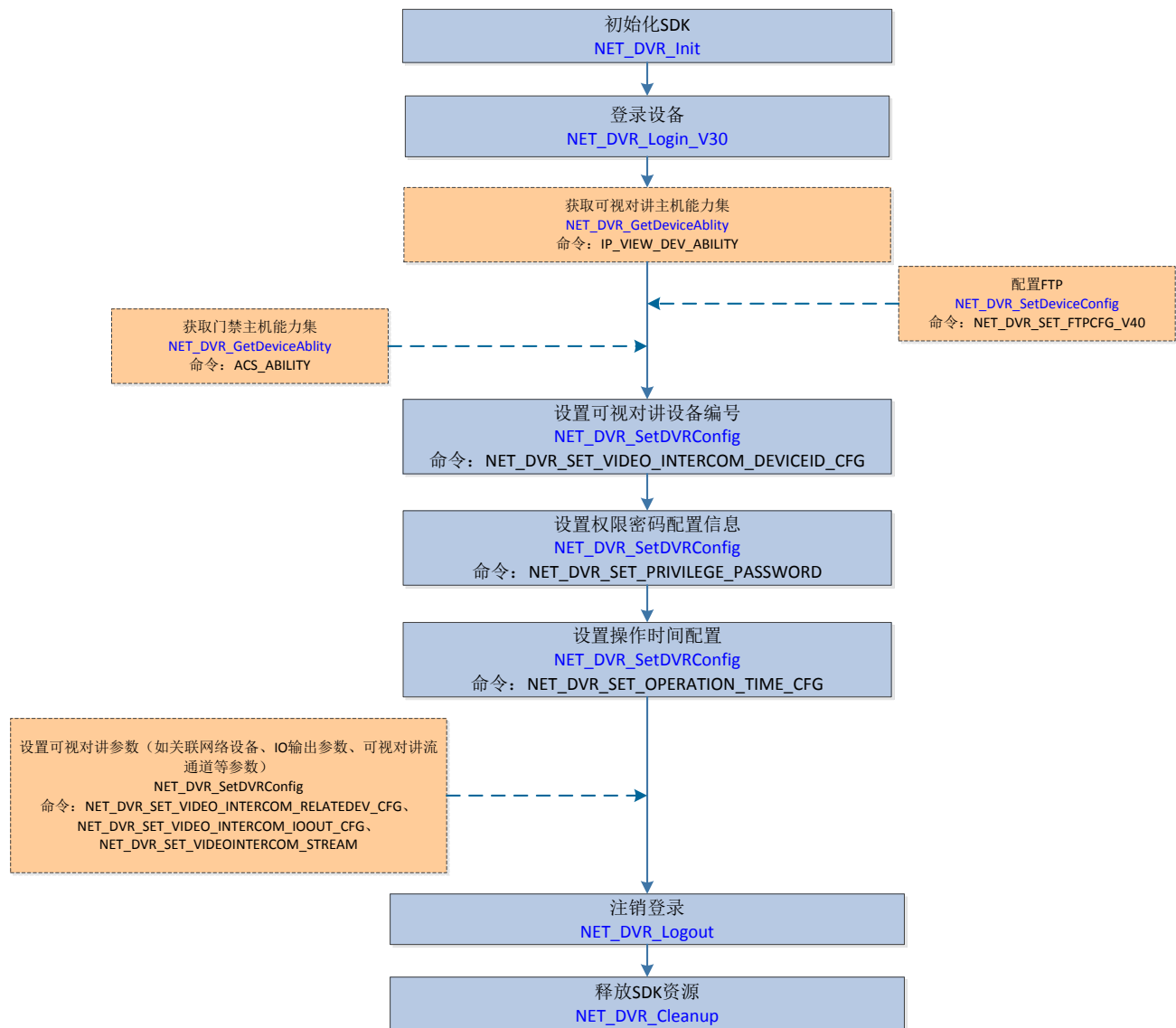
- 设置可视对讲流通道参数，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_VIDEOINTERCOM_STREAM`），可以设置可视对讲设备编号（长号）、设备名称、视频源类型等参数；
- 设置可视对讲 IO 输出参数，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_VIDEO_INTERCOM_IOOUT_CFG`），可以设置 IO 输出用途；
- 设置梯控器参数，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_ELEVATORCONTROL_CFG_V40`），可以设置是否启用梯控、接口类型、RS485 协议类型、网口协议类型、梯控服务器端口以及梯控服务器 IP 地址等参数。
- 设置宽动态参数，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_WDR_CFG`），可以设置宽动态相关参数。
- 设置按键呼叫住户配置参数，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_CALL_ROOM_CFG`），可以设置按键呼叫住户配置参数。
- 设置音频输出口音量调节参数，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_AUDIOOUT_VOLUME_CFG`），可以设置音频输出口音量调节参数。
- 查询门口机注册信息，通过接口 `NET_DVR_StartRemoteConfig` 获取门口机注册信息，然后调用 `NET_DVR_GetNextRemoteConfig` 接口逐个获取查找到的结果信息，调用接口 `NET_DVR_StopRemoteConfig` 关闭长连接。
- 远程开锁，通过接口 `NET_DVR_RemoteControl`（命令：`NET_DVR_REMOTECONTROL_GATEWAY`）实现。
- 公告信息下发，通过接口 `NET_DVR_RemoteControl`（命令：`NET_DVR_REMOTECONTROL_NOTICE_DATA`）实现。
- 设置 RS485 参数，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_ALARM_RS485CFG`），设置 485 参数。
- 设置防区参数，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_ALARMIN_PARAM`），设置防区参数。
- 设置门参数，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_DOOR_CFG`），设置门参数。
- 设置卡参数配置，通过接口 `NET_DVR_StartRemoteConfig`（命令：`NET_DVR_SET_CARD_CFG`），设置卡参数；通过接口 `NET_DVR_SendRemoteConfig`（命令：`NET_DVR_SET_CARD_CFG`）设置，接口 `NET_DVR_StopRemoteConfig` 关闭长连接。
- 清空门禁卡信息调用接口 `NET_DVR_RemoteControl`（命令：`NET_DVR_CLEAR_ACS_PARAM`）。
- 预览部分调用 `NET_DVR_RealPlay_V40` 接口开始预览，停止预览调用接口 `NET_DVR_StopRealPlay`，具体流程参考预览接口流程；
- 语音对讲调用 `NET_DVR_StartVoiceCom_V30` 开始语音对讲，`NET_DVR_SetVoiceComClientVolume` 设置语音对讲客户端音量，`NET_DVR_StopVoiceCom` 停止语音对讲，具体流程参考语音对讲接口流程；
- 报警可以调用接口 `NET_DVR_SetDVRMessageCallBack_V30` 报警消息回调获取到相关报警信息，其中报警类型是 `COMM_UPLOAD_VIDEO_INTERCOM_EVENT` 和 `COMM_ALARM_VIDEO_INTERCOM`，可以通过布防 `NET_DVR_SetupAlarmChan_V41` 和监听 `NET_DVR_StartListen_V30` 的方式获取到报警。具体流程请参考报

警模块流程。

3.2 楼宇可视对讲功能接口流程

3.2.1 楼宇可视对讲远程参数配置

图 3.2 楼宇可视对讲远程参数配置接口调用流程



- 登录设备后通过接口 [NET_DVR_GetDeviceAbility](#)（能力集类型：IP_VIEW_DEV_ABILITY）获取可视对讲能力集。如是否支持自动注册、本地能力、支持的音频编码类型、呼叫能力、门口机、管理机、别墅门口机、二次确认机、权限密码、操作时间。
- 设置 FTP 信息，通过接口 [NET_DVR_SetDeviceConfig](#)（命令：NET_DVR_SET_FTPCFG_V40），设置上传的 FTP 参数信息。
- 设置流的来源信息，通过接口 [NET_DVR_SetDeviceConfig](#)（命令：NET_DVR_SET_STREAM_SRC_INFO），

设置流的来源参数。

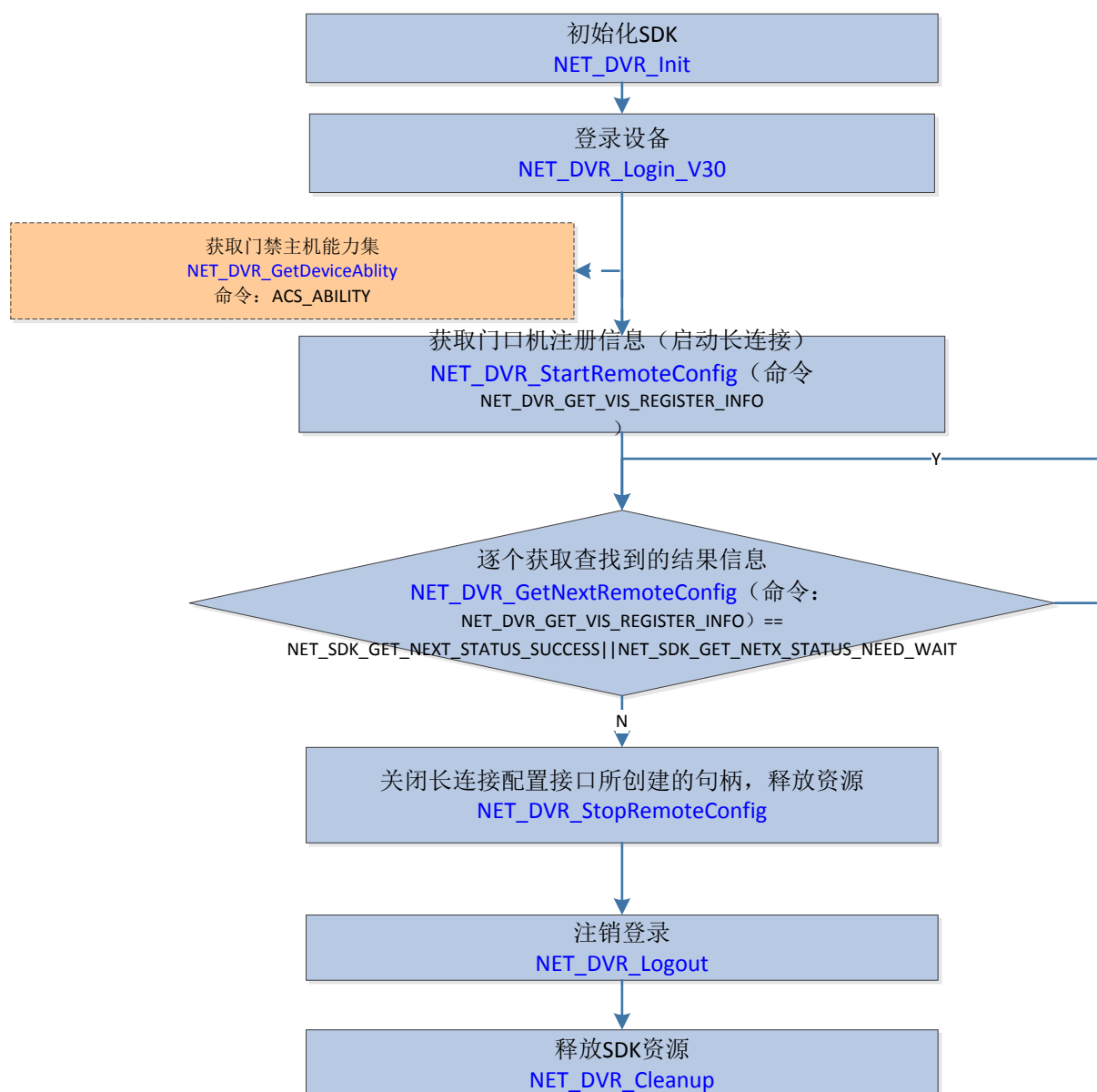
- 设置可视对讲设备编号，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_VIDEO_INTERCOM_DEVICEID_CFG`），设置设备类型、是否自动注册以及设备编号参数。
- 设置操作时间配置，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_OPERATION_TIME_CFG`），设置管理及或门口机/围墙机/别墅门口机/二次确认及操作时间配置参数。
- 设置关联网络设备参数，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_VIDEO_INTERCOM_RELATEDDEV_CFG`），设置门口机/围墙机关联网络设备参数配置或者管理机关联网络，别墅门口机关联网络、二次确认机关联网络设备参数配置。
- 设置可视对讲流通道参数，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_VIDEOINTERCOM_STREAM`），可以设置可视对讲设备编号（长号）、设备名称、视频源类型等参数；
- 设置可视对讲 IO 输出参数，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_VIDEO_INTERCOM_IOOUT_CFG`），可以设置 IO 输出用途；
- 设置梯控器参数，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_ELEVATORCONTROL_CFG_V40`），可以设置是否启用梯控、接口类型、RS485 协议类型、网口协议类型、梯控服务器端口以及梯控服务器 IP 地址等参数。
- 设置宽动态参数，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_WDR_CFG`），可以设置宽动态相关参数。
- 设置按键呼叫住户配置参数，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_CALL_ROOM_CFG`），可以设置按键呼叫住户配置参数。
- 设置音频输出口音量调节参数，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_AUDIOOUT_VOLUME_CFG`），可以设置音频输出口音量调节参数。
- 查询门口机注册信息，通过接口 `NET_DVR_StartRemoteConfig` 获取门口机注册信息，然后调用 `NET_DVR_GetNextRemoteConfig` 接口逐个获取查找到的结果信息，调用接口 `NET_DVR_StopRemoteConfig` 关闭长连接。
- 远程开锁，通过接口 `NET_DVR_RemoteControl`（命令：`NET_DVR_REMOTECONTROL_GATEWAY`）实现。
- 公告信息下发，通过接口 `NET_DVR_RemoteControl`（命令：`NET_DVR_REMOTECONTROL_NOTICE_DATA`）实现。
- 设置 RS485 参数，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_ALARM_RS485CFG`），设置 485 参数。
- 设置防区参数，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_ALARMIN_PARAM`），设置防区参数。
- 设置门参数，通过接口 `NET_DVR_SetDVRConfig`（命令：`NET_DVR_SET_DOOR_CFG`），设置门参数。
- 设置卡参数配置，通过接口 `NET_DVR_StartRemoteConfig`（命令：`NET_DVR_SET_CARD_CFG`），设置卡参数；通过接口 `NET_DVR_SendRemoteCofig`（命令：`NET_DVR_SET_CARD_CFG`）设置，接口 `NET_DVR_StopRemoteConfig` 关闭长连接。

- 清空门禁卡信息调用接口 `NET_DVR_RemoteControl`（命令：`NET_DVR_CLEAR_ACS_PARAM`）。

[调用实例代码](#)

3.2.2 查询门口机注册信息

图 3.3 查询门口机注册信息



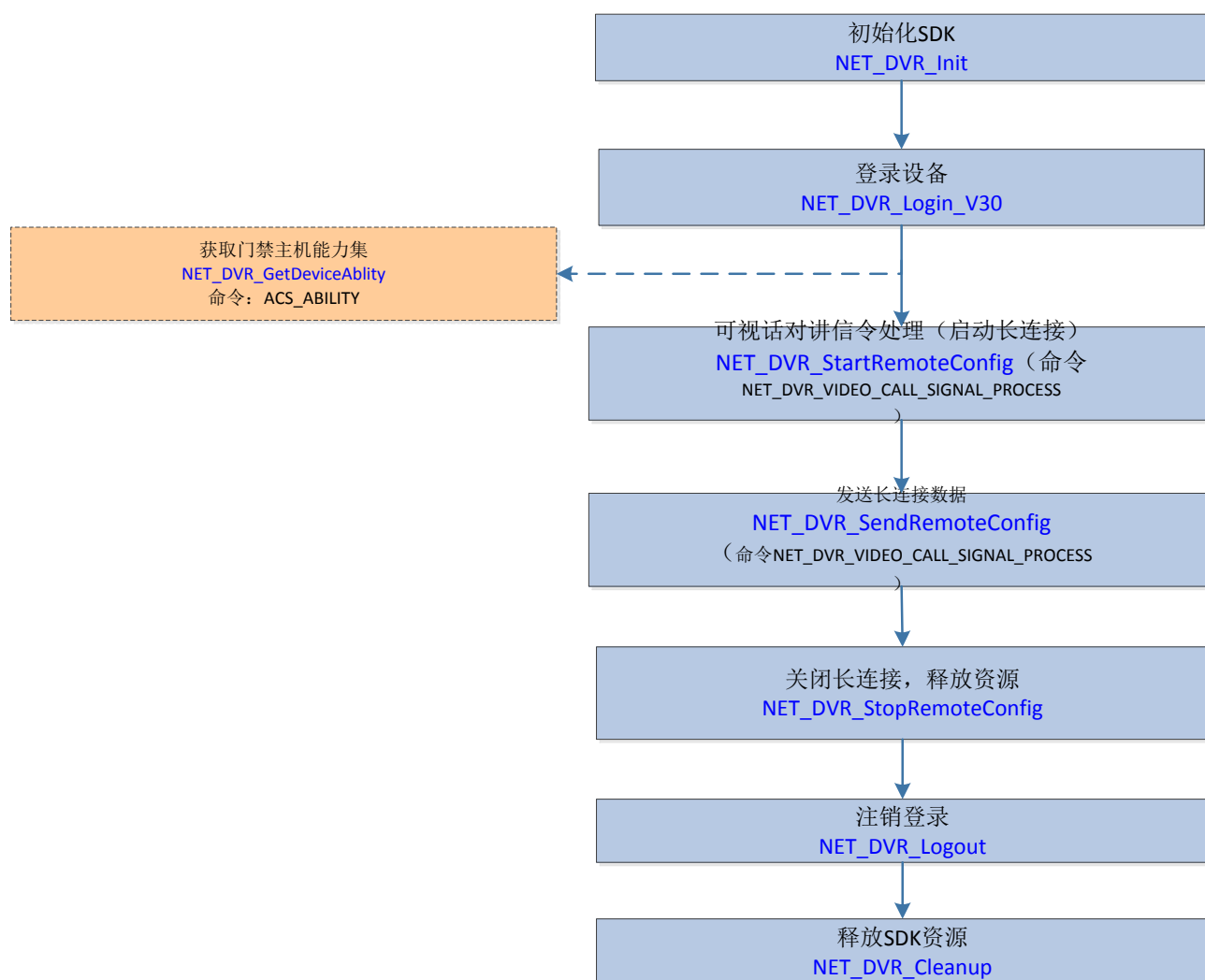
- 设备登录后调用接口 `NET_DVR_StartRemoteConfig` 启用长连接，通过命令 `NET_DVR_GET_VIS_REGISTER_INFO`，获取门口机的注册信息。
- 调用接口 `NET_DVR_GetNextRemoteConfig`，通过命令 `NET_DVR_GET_VIS_REGISTER_INFO`，通过返回的结构体 `NET_DVR_VIS_REGISTER_INFO` 逐条获取。其中可通过接口返回值判断当前查询的状态，若为 `NET_SDK_GET_NEXT_STATUS_SUCCESS` 和 `NET_SDK_GET_NETX_STATUS_NEED_WAIT` 则需要继续调用

NET_DVR_GetNextRemoteConfig 继续获取。

- 结束则调用接口 NET_DVR_StopRemoteConfig 接口关闭长连接，释放资源。

3.3 可视通话信令交互

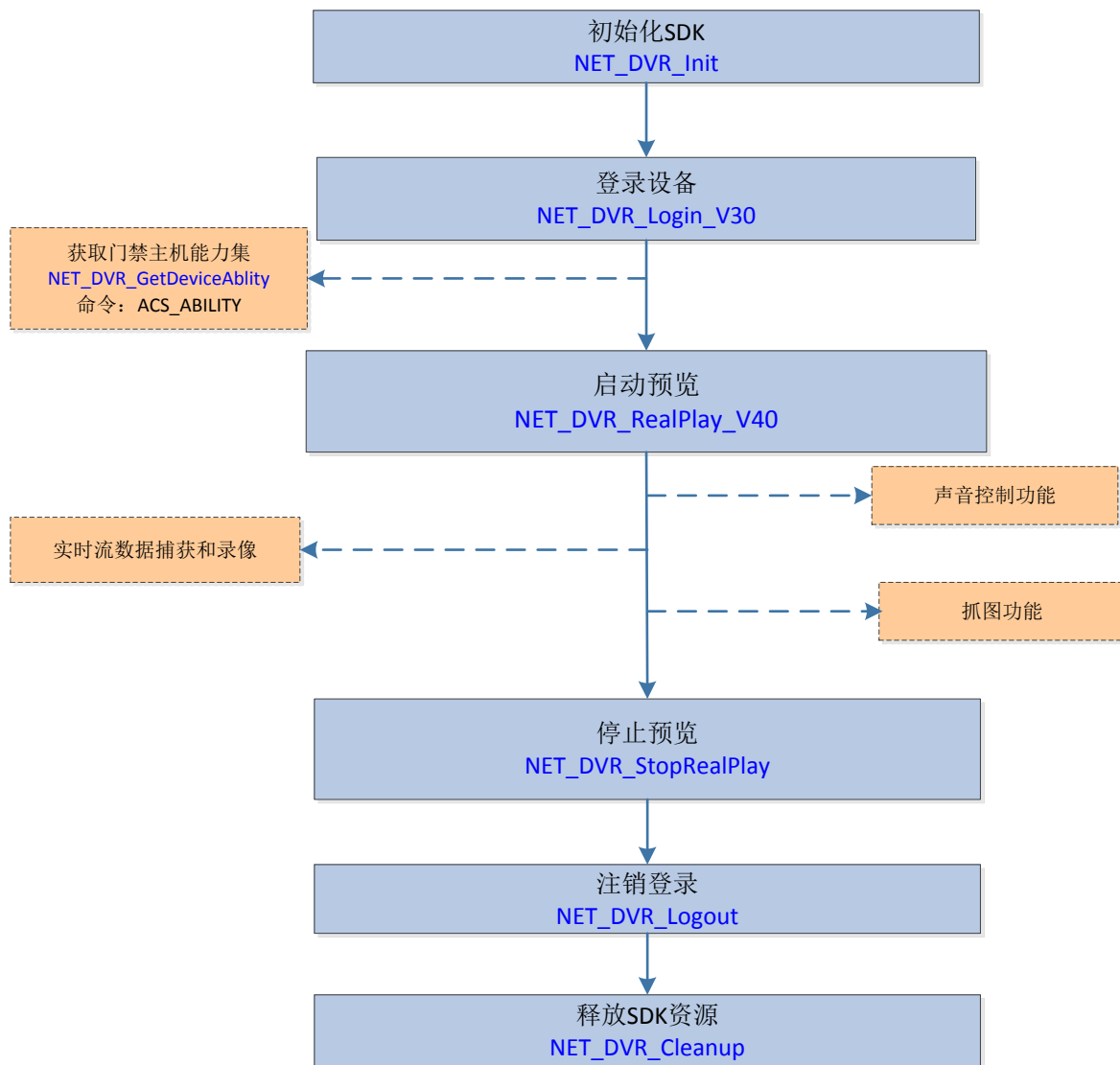
图 3.4 可视通话信令交互



- 设备登录后调用接口 NET_DVR_StartRemoteConfig 启用长连接，通过命令 NET_DVR_VIDEO_CALL_SIGNAL_PROCESS，可视话对讲信令处理。
- 调用接口 NET_DVR_SendRemoteConfig，若 NET_DVR_StartRemoteConfig 接口中的 dwCommand 为 NET_DVR_VIDEO_CALL_SIGNAL_PROCESS，则对应 NET_DVR_SendRemoteConfig 接口的 dwDataType 为 0。
- 结束则调用接口 NET_DVR_StopRemoteConfig 接口关闭长连接，释放资源。

3.4 预览模块

图 3.5 预览模块



- 设备登录后调用接口 `NET_DVR_RealPlay_V40` 接口实现预览，接口 `NET_DVR_StopRealPlay` 接口停止预览。
 - 声音控制功能主要实现独占、共享声音的打开和关闭；音量的控制。相关接口有：[NET_DVR_OpenSound](#)、[NET_DVR_CloseSound](#)、[NET_DVR_OpenSoundShare](#)、[NET_DVR_CloseSoundShare](#)、[NET_DVR_Volume](#) 等。
 - 实时流数据捕获和录像模块主要实现数据回调和本地录像的功能，可以供用户后续处理。相关接口有：[NET_DVR_SetRealDataCallBack](#)、[NET_DVR_SetStandardDataCallBack](#)、[NET_DVR_SaveRealData](#) 等。
 - 抓图功能主要实现对当前解码图像的捕获，保存格式为 BMP。相关接口有：[NET_DVR_CapturePicture](#)。

3.5 语音对讲模块

图 3.6 语音对讲模块

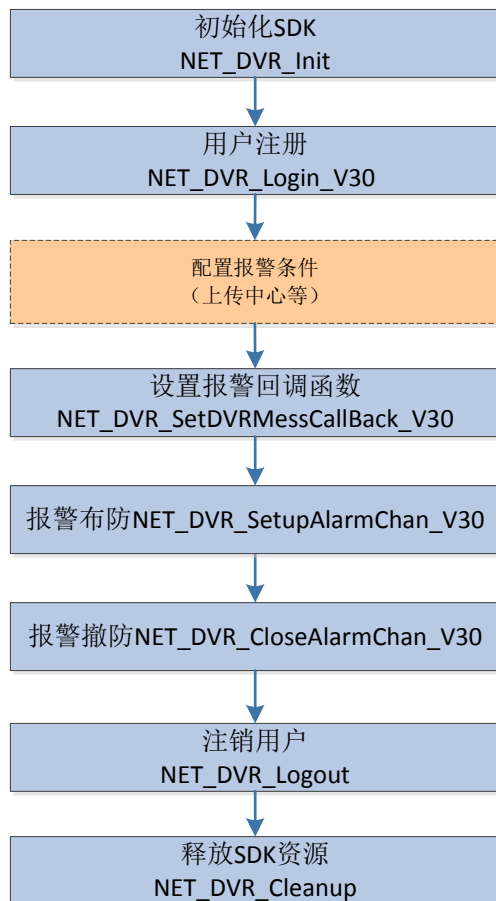


- 语音对讲功能实现 PC 机与设备间音频的发送和接收。在成功注册设备后调用 [NET_DVR_StartVoiceCom_V30](#) 接口完成，同时在该接口中用户可以通过设置回调函数获取当前设备发送或者 PC 机采集的数据（按需要选择回调编码后或者 PCM 数据）。
- Windows 64 位或者 Linux 系统下不支持语音对讲。

3.6 报警模块流程

3.6.1 报警布防

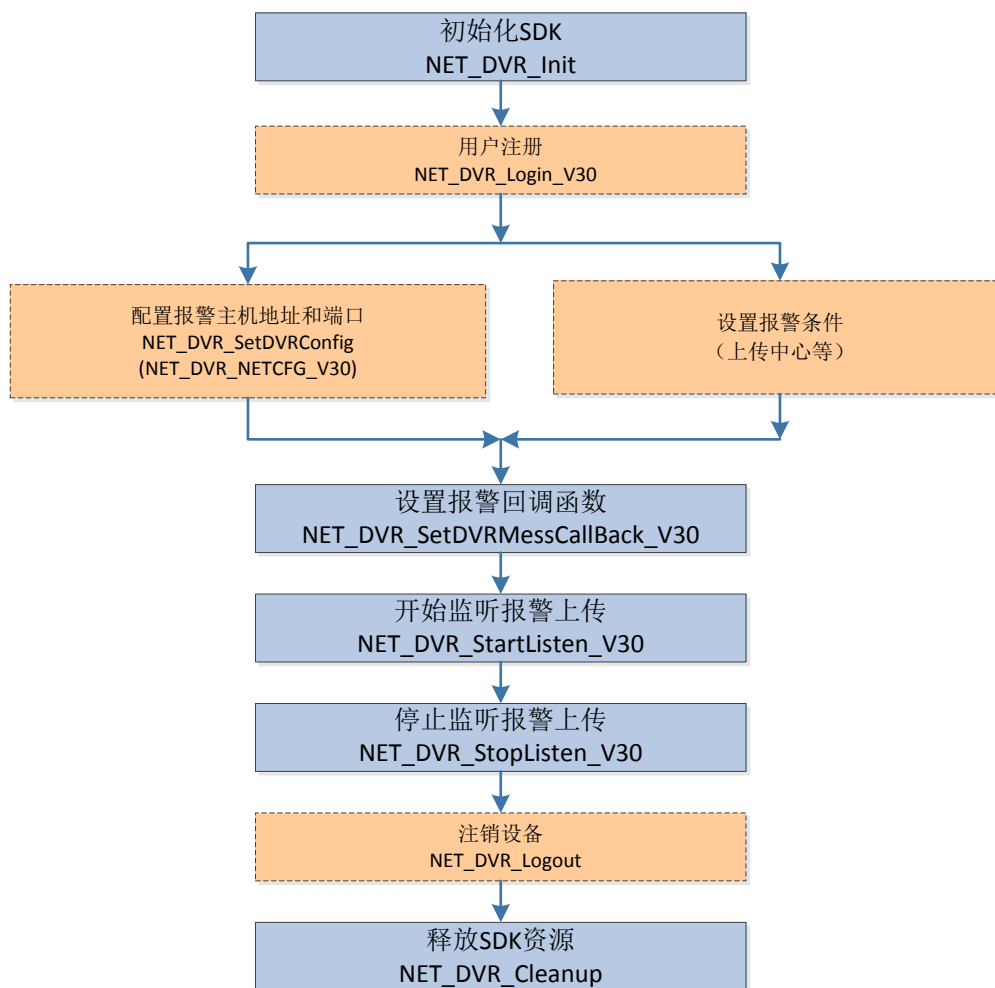
图 3.7 报警布防调用流程



- “布防”报警方式是指 SDK 主动连接设备，并发起报警上传命令，设备发生报警立即发送给 SDK。
- “布防”方式需要先进行用户注册（[NET DVR Login V30](#)），然后需要设置报警回调函数（[NET DVR SetDVRMessageCallBack V30](#)）和对设备进行布防（[NET DVR SetupAlarmChan V41](#)）。报警信息在设置的报警回调函数中获取。整个报警上传过程结束后还需要调用撤防接口等操作，释放相关资源。
- 虚线框部分是实现报警信息上传的必要条件，主要完成相关的报警条件和处理方法（须设置上传中心）的配置。比如，报警输入参数配置，相关的接口：[NET DVR GetDVRConfig](#)（命令：[NET_DVR_GET_ALARMINCFG_V40](#)）、[NET DVR SetDVRConfig](#)（命令：[NET_DVR_SET_ALARMINCFG_V40](#)）。相关参数如果已经配置完成，那么虚线框部分可以省略。

3.6.2 报警监听

图 3.8 报警监听调用流程



- “监听”报警方式：是指 SDK 不主动发起连接设备，只是在设定的端口上监听接收设备主动上传的报警信息。
- 这个过程需要远程配置设备的报警主机地址（即 PC 机地址）和报警主机端口（即 PC 的监听端口），报警主机就在该端口上监听接收设备主动上传的报警信息。如果报警主机地址和报警主机端口已配置完成，那么“报警（监听）的流程图”中虚线框“用户注册”和“配置报警主机地址和端口”部分就可以省略，但事先没有配置，就必须调用参数配置接口（[NET_DVR_GetDVRConfig](#) 和 [NET_DVR_SetDVRConfig](#)）对设备的网络参数（[NET_DVR_NETCFG_V30](#)）进行配置。而虚线框“配置报警条件和处理方法”部分与“布防”中的一致。对以上需要配置的参数都设置完后，调用 [NET_DVR_StartListen_V30](#) 函数，开启 SDK 的监听端口，准备接收设备上传的报警信息。
- 该方式适用于多个设备向一台客户端上传报警，而且不需要设备登录即可完成，设备重启后不影响报警上传；缺点是设备只支持一个报警主机地址和端口号的配置。
- 报警可以调用接口 [NET_DVR_SetDVRMessageCallBack_V30](#) 报警消息回调获取到相关报警信息，其中报警类型是 [COMM_UPLOAD_VIDEO_INTERCOM_EVENT](#) 和 [COMM_ALARM_VIDEO_INTERCOM](#)，可以通过监听方式获取到报警。

4 函数调用实例

4.1 楼宇可视对讲示例代码

[相关模块流程图](#)

```
#include <stdio.h>
#include <iostream>
#include "Windows.h"
#include "HCNetSDK.h"
using namespace std;

void CALLBACK MessageCallback(LONG lCommand, NET_DVR_ALARMER *pAlarmer, char *pAlarmInfo, DWORD dwBufLen, void*
pUser)
{
    int i;
    NET_DVR_ALARMINFO struAlarmInfo;
    memcpy(&struAlarmInfo, pAlarmInfo, sizeof(NET_DVR_ALARMINFO));
    switch(lCommand)
    {
        case COMM_UPLOAD_VIDEO_INTERCOM_EVENT: // 可视对讲事件记录信息
            LPNET_DVR_VIDEO_INTERCOM_EVENT pVideoIntercomEvent = (LPNET_DVR_VIDEO_INTERCOM_EVENT)pAlarmMsg;
            if (pVideoIntercomEvent->byEventType == 1 &&
                pVideoIntercomEvent->uEventInfo.struUnlockRecord.dwPicDataLen > 0) // 开锁记录
            {
                //保持开锁图片
                pVideoIntercomEvent->uEventInfo.struUnlockRecord.plImage = new
                BYTE[pVideoIntercomEvent->uEventInfo.struUnlockRecord.dwPicDataLen];
                memcpy(pVideoIntercomEvent->uEventInfo.struUnlockRecord.plImage, ((LPNET_DVR_VIDEO_INTERCOM_EVENT)pAlarm
                Info)->uEventInfo.struUnlockRecord.plImage,
                    pVideoIntercomEvent->uEventInfo.struUnlockRecord.dwPicDataLen);
            }
            break;
        case COMM_ALARM_VIDEO_INTERCOM: // 可视对讲报警信息
            char szInfoBuf[1024] = {0};
            LPLOCAL_ALARM_INFO pAlarmDev = (LPLOCAL_ALARM_INFO)(wParam);
            char *pAlarmInfo = (char *) (lParam);
            int iWinIndex = -1;
            int i = 0;
            int iDeviceIndex = pAlarmDev->iDeviceIndex;

            NET_DVR_VIDEO_INTERCOM_ALARM struVideoIntercomAlarm = {0};
            CTime time = CTime::GetCurrentTime();
```

```

memcpy(&struVideoIntercomAlarm, pAlarmInfo, sizeof(struVideoIntercomAlarm));

if (struVideoIntercomAlarm.byAlarmType == 1)// 防区报警
{
    sprintf(szInfoBuf, "Video Intercom Alarm Time[%4.4d%2.2d%2.2d%2.2d%2.2d] DevNumber[%s]
AlarmType[%d] ZoneName[%s] ZoneIndex[%d] ZoneType[%d]",
            struVideoIntercomAlarm.struTime.wYear, struVideoIntercomAlarm.struTime.byMonth,
struVideoIntercomAlarm.struTime.byDay,
            struVideoIntercomAlarm.struTime.byHour, struVideoIntercomAlarm.struTime.byMinute,
struVideoIntercomAlarm.struTime.bySecond,

            struVideoIntercomAlarm.byDevNumber, struVideoIntercomAlarm.byAlarmType, struVideoIntercomAlarm.uAlarmInfo.str
uZoneAlarm.byZoneName,

            struVideoIntercomAlarm.uAlarmInfo.struZoneAlarm.dwZonendex, struVideoIntercomAlarm.uAlarmInfo.struZoneAlarm.b
yZoneType);
}
Else//其他报警
{
    sprintf(szInfoBuf, "Video Intercom Alarm Time[%4.4d%2.2d%2.2d%2.2d%2.2d] DevNumber[%s]
AlarmType[%d]",
            struVideoIntercomAlarm.struTime.wYear, struVideoIntercomAlarm.struTime.byMonth,
struVideoIntercomAlarm.struTime.byDay,
            struVideoIntercomAlarm.struTime.byHour, struVideoIntercomAlarm.struTime.byMinute,
struVideoIntercomAlarm.struTime.bySecond,
            struVideoIntercomAlarm.byDevNumber, struVideoIntercomAlarm.byAlarmType);

    }
    break;
default:
    break;
}
}

void main() {
    //-----
    // 初始化
    NET_DVR_Init();
    //设置连接时间与重连时间
    NET_DVR_SetConnectTime(2000, 1);
    NET_DVR_SetReconnect(10000, true);

    //-----
    // 注册设备
    LONG IUserID;

```

```
NET_DVR_DEVICEINFO_V30 struDeviceInfo;
IUserID = NET_DVR_Login_V30("172.0.0.100", 8000, "admin", "12345", &struDeviceInfo);
if (IUserID < 0)
{
    printf("Login error, %d\n", NET_DVR_GetLastError());
    NET_DVR_Cleanup();
    return;
}

//设置报警回调函数
NET_DVR_SetDVRMessageCallBack_V30(MessageCallback, NULL);

//启用监听
LONG IHandle;
IHandle = NET_DVR_StartListen_V30(NULL, 7200, MessageCallback, NULL);
if (IHandle < 0)
{
    printf("NET_DVR_StartListen_V30 error, %d\n", NET_DVR_GetLastError());
    NET_DVR_Logout(IUserID);
    NET_DVR_Cleanup();
    return;
}

Sleep(5000);
//停止监听
if (!NET_DVR_StopListen_V30(IHandle))
{
    printf("NET_DVR_StopListen_V30 error, %d\n", NET_DVR_GetLastError());
    NET_DVR_Logout(IUserID);
    NET_DVR_Cleanup();
    return;
}

//注销用户
NET_DVR_Logout(IUserID);
//释放 SDK 资源
NET_DVR_Cleanup();
return;
}
```

5 函数说明

5.1 SDK 初始化

5.1.1 初始化 SDK **NET_DVR_Init**

函 数: BOOL NET_DVR_Init()

参 数: 无

返回值: TRUE 表示成功, FALSE 表示失败。

说 明: 调用设备网络 SDK 其他函数的前提

[返回目录](#)

5.1.2 释放 SDK 资源 **NET_DVR_Cleanup**

函 数: BOOL NET_DVR_Cleanup()

参 数: 无

返回值: TRUE 表示成功, FALSE 表示失败。

说 明: 在结束之前最后调用。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

[返回目录](#)

5.1.3 设置网络连接超时时间和连接尝试次数 **NET_DVR_SetConnectTime**

函 数: BOOL NET_DVR_SetConnectTime(DWORD dwWaitTime, DWORD dwTryTime)

参 数: [in] dwWaitTime 超时时间, 单位毫秒, 取值范围[300,75000], 实际最大超时时间因系统的 connect 超时时间而不同。

[in] dwTryTimes 连接尝试次数 (保留)

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: **SDK 默认建立连接的超时时间为 3 秒。** SDK4.0 及以后版本中当设置的超时时间超过或低于限制的值时接口不返回失败, 将取最接近的上下限限制值作为实际的超时时间。

[返回目录](#)

5.1.4 设置重连功能 **NET_DVR_SetReconnect**

函 数: BOOL NET_DVR_SetReconnect (DWORD dwInterval, BOOL bEnableRecon)

参 数: [in] dwInterval 重连间隔, 单位:毫秒

[in] bEnableRecon 是否重连, 0-不重连, 1-重连, 参数默认值为 1

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通

过错误码判断出错原因。

说明： 该接口可以同时控制预览、透明通道和布防的重连功能。不调用该接口时，SDK 默认启动预览、透明通道和布防的重连功能，重连时间间隔为 5 秒。

[返回目录](#)

5.1.5 设置接收超时时间 **NET_DVR_SetRecvTimeOut**

函数： BOOL NET_DVR_SetRecvTimeOut(DWORD nRecvTimeOut)

参数： [in] nRecvTimeOut 接收超时时间，单位毫秒，默认为 5000，最小为 3000 毫秒

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明： 该接口用于设置接收超时时间，例如预览接收实时流数据、回放下载接收录像数据、报警接收报警信息等接收超时时间。

[返回目录](#)

5.1.6 获取所有 IP，用于支持多网卡接口 **NET_DVR_GetLocalIP**

函数： BOOL NET_DVR_GetLocalIP(char strIP[16][16], DWORD *pValidNum, BOOL *pEnableBind)

参数： [out] strIP 存放 IP 的缓冲区，不能为空

[out] pValidNum 所有有效 IP 的数量

[out] pEnableBind 是否绑定

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明： 该接口获取客户端本地多网卡的所有 IP 地址，可以通过接口 [NET_DVR_SetValidIP](#) 选择要使用的 IP 地址。

[返回目录](#)

5.1.7 设置 IP 绑定 **NET_DVR_SetValidIP**

函数： BOOL NET_DVR_SetValidIP(DWORD dwIPIndex, BOOL bEnableBind)

参数： [in] dwIPIndex 选择使用的 IP 下标，由 [NET_DVR_GetLocalIP](#) 获取

[in] bEnableBind 是否绑定

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明：

[返回目录](#)

5.1.8 通过解析服务器获取设备的动态 IP 地址和端口号

NET_DVR_GetDVRIPByResolveSvr_EX

函数： BOOL NET_DVR_GetDVRIPByResolveSvr_EX(char* sServerIP, WORD wServerPort, BYTE* sDVRName,

WORD wDVRNameLen, BYTE* sDVRSerialNumber, WORD wDVRSerialLen, char* sGetIP, DWORD* dwPort)

参 数: [in]sServerIP 解析服务器的 IP 地址或域名
[in]wServerPort 解析服务器的端口号, IP Server 解析服务器端口号为 7071, HiDDNS 服务器的端口号为 80
[in]sDVRName 设备名称或者设备域名
[in]wDVRNameLen 设备名称(域名)的长度
[in]sDVRSerialNumber 设备的序列号
[in]wDVRSerialLen 设备序列号的长度
[out]sGetIP 获取到的设备 IP 地址指针
[out]dwPort 获取到的设备端口号指针

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 该接口中的设备名称和设备序列号不能同时为空。支持的解析服务器有 IPServer 和 hiDDNS。通过设备域名或者序列号解析出设备当前 IP 地址和端口, 然后调用 [NET_DVR_Login_V30](#) 登录设备。

[返回目录](#)

5.2 实时预览

5.2.1 设置播放显示模式 **NET_DVR_SetShowMode**

函 数: BOOL NET_DVR_SetShowMode(DWORD dwShowType, COLORREF colorKey)

参 数: [in] DWORD 显示模式(linux 下只自持 NORMALMODE 模式)

```
enum{
    NORMALMODE = 0,
    OVERLAYMODE
}
```

[in] IChannel 用户设置的透明色, 在 OVERLAY 模式下需要设置, colorKey 是一个 32 位的值 0x00bbggrr, 最高字节为 0, 后三个字节分别表示 b、g、r 的值

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

- 设置播放器显示模式, 需在预览之前设置。NORMALMODE 模式可以同时显示多窗口, 但是对显卡有一定的要求; OVERLAYMODE 模式只能同时显示一个窗口, 但是对显卡基本没有要求。
- 透明色相当于一层透视膜, 显示的画面只能穿过这种颜色, 而其他的颜色将挡住显示的画面, 用户应该在显示窗口中涂上这种颜色才能看到显示画面, 一般应该使用一种不常用的颜色作为透明色, colorKey 是一个 32 位的值 0x00bbggrr, 最高字节为 0, 后三个字节分别表示 b、g、r 的值。

播放器有两种显示模式: 普通模式和 OVERLAY 方式, 使用 OVERLAY 模式的优点是: 大部分显卡都支持 OVERLAY, 在一些不支持 BLT 硬件缩放和颜色转换的显卡上(如 SIS

系列显卡)使用 OVERLAY 模式,可以大大降低 CPU 利用率并提高画面质量(相对于软件实现缩放、颜色转换)。缺点是:同时只能播放一路图象,不能实现大规模集中监控。

在一块显卡中同一时刻只能有一个 OVERLAY 表面处于活动状态,如果此时系统中已经有程序使用了 OVERLAY,那么播放器就不能再创建 OVERLAY 表面,它将自动改成普通的模式,并不返回 FALSE,一些常用的播放器,例如我们卡的预览都可能使用了 OVERLAY 表面,同样,如果我们的 SDK 中使用了 OVERLAY 表面,那么其他的程序将不能再使用 OVERLAY 表面。

- V7.x 版本的播放库默认采用 d3d 的显示方式, V6.x 版本的播放库采用 ddraw 的显示方式。Overlay 模式只有在 ddraw 模式下才有,因此现在播放库不支持 overlay 模式,调用 NET_DVR_SetShowMode 设置 overlay 无效。

[返回目录](#)

5.2.2 主码流动态产生一个关键帧 **NET_DVR_MakeKeyFrame**

函 数: BOOL NET_DVR_MakeKeyFrame(LONG IUserID, DWORD IChannel)

参 数: [in] IUserID NET_DVR_Login_V40 的返回值
[in] IChannel 通道号

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码,通过错误码判断出错原因。

说 明: 此接口用于重置 I 帧,根据设置的预览参数([NET_DVR_CLIENTINFO](#))为主码流或者子码流分别调用 NET_DVR_MakeKeyFrame 或者 [NET_DVR_MakeKeyFrameSub](#) 实现重置 I 帧。

[返回目录](#)

5.2.3 子码流动态产生一个关键帧 **NET_DVR_MakeKeyFrameSub**

函 数: BOOL NET_DVR_MakeKeyFrameSub(LONG IUserID, DWORD IChannel)

参 数: [in] IUserID NET_DVR_Login_V40 的返回值
[in] IChannel 通道号

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码,通过错误码判断出错原因。

说 明: 此接口用于重置 I 帧,根据设置的预览参数([NET_DVR_CLIENTINFO](#))为主码流或者子码流分别调用 [NET_DVR_MakeKeyFrame](#) 或者该接口实现重置 I 帧。

[返回目录](#)

5.2.4 实时预览 **NET_DVR_RealPlay_V40**

函 数: LONG NET_DVR_RealPlay_V40(LONG IUserID, LPNET_DVR_PREVIEWINFO lpPreviewInfo, REALDATACALLBACK fRealDataCallBack_V30, void *pUser)

参 数: [in] IUserID NET_DVR_Login_V40 的返回值

[in] lpPreviewInfo	预览参数，包括码流类型、取流协议、通道号、预览窗口句柄等，详见结构体： NET_DVR_PREVIEWINFO
[in] fRealDataCallBack_V30	码流数据回调函数
[in] pUser	用户数据

```
typedef void(CALLBACK *REALDATACALLBACK) (LONG lRealHandle, DWORD dwDataType, BYTE *pBuffer, DWORD dwBufSize, void *pUser);
```

[out]lRealHandle	当前的预览句柄
[out]dwDataType	数据类型，详见表 5.1
[out] pBuffer	存放数据的缓冲区指针
[out]dwBufSize	缓冲区大小
[out]pUser	用户数据

表 5.1 码流数据类型

dwDataType 宏定义	宏定义值	含义
NET_DVR_SYSHEAD	1	系统头数据
NET_DVR_STREAMDATA	2	流数据（包括复合流或音视频分开的视频流数据）
NET_DVR_AUDIOSTREAMDATA	3	音频数据

返回值：-1 表示失败，其他值作为 NET_DVR_StopRealPlay 等函数的句柄参数。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

- 说明：**
- 该接口预览参数结构中可以设置当前预览操作是否阻塞（通过 bBlocked 参数设置），若设为不阻塞，表示发起与设备的连接就认为连接成功，如果发生码流接收失败、播放失败等情况以预览异常的方式通知上层。在循环播放的时候可以减短停顿的时间，与 NET_DVR_RealPlay 处理一致。若设为阻塞，表示直到播放操作完成才返回成功与否。
 - 该接口中的回调函数可以置为空，这样该函数将不回调码流数据给用户，不过用户仍可以通过接口 [NET_DVR_SetRealDataCallBack](#) 或 [NET_DVR_SetStandardDataCallBack](#) 注册捕获码流数据的回调函数以捕获码流数据。
 - fRealDataCallBack_V30 回调函数中不能执行可能会占用时间较长的接口或操作，不建议调用该 SDK（HCNetSDK.dll）本身的接口。
 - Linux 64 位系统不支持软解码功能，因此需要将窗口句柄传 NULL，设置回调函数，只取流不解码显示。
 - 客户端异常离线时，设备端对取流连接的保持时间为 10 秒。

[返回目录](#)

5.2.5 停止预览 **NET_DVR_StopRealPlay**

函数：LONG NET_DVR_StopRealPlay (LONG lRealHandle)

参数：[in] lRealHandle 预览句柄，NET_DVR_RealPlay_V40 的返回值

返回值：TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明：

[返回目录](#)

5.2.6 获取预览时用来解码和显示的播放库句柄 **NET_DVR_GetRealPlayerIndex**

函 数： int NET_DVR_GetRealPlayerIndex(LONG IRealHandle)

参 数： [in] IRealHandle 预览句柄，NET_DVR_RealPlay_V40 的返回值

返回值： -1 表示失败，其他值表示播放句柄。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说 明： 用户可以通过返回的句柄自行实现播放库 SDK 提供的其他功能，详见本公司提供的软解码库函数说明《播放器 SDK 编程指南》。例如使用 PlayM4_GetBMP(LONG nPort,.....)、PlayM4_GetJPEG(LONG nPort,.....)这两个接口时，即可实现将当前预览图像以 BMP 或 JPEG 格式抓图保存到内存中： PlayM4_GetBMP(NET_DVR_GetRealPlayerIndex(),.....) PlayM4_GetJPEG(NET_DVR_GetRealPlayerIndex(),.....)

[返回目录](#)

5.3 预览显示视频参数配置

5.3.1 获取预览视频显示参数 **NET_DVR_ClientGetVideoEffect**

函 数： BOOL NET_DVR_ClientGetVideoEffect(LONG IRealHandle,DWORD *pBrightValue, DWORD *pContrastValue,DWORD *pSaturationValue,DWORD *pHueValue)

参 数： [in]IRealHandle NET_DVR_RealPlay_V40 的返回值
[out]pBrightValue 亮度指针，取值范围[1,10]
[out]pContrastValue 对比度指针，取值范围[1,10]
[out]pSaturationValue 饱和度指针，取值范围[1,10]
[out]pHueValue 色度指针，取值范围[1,10]

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说 明： 需要预览才能获取视频参数。

[返回目录](#)

5.3.2 获取预览视频显示参数 **NET_DVR_GetVideoEffect**

函 数： BOOL NET_DVR_GetVideoEffect(LONG IUserID, LONG IChannel,DWORD *pBrightValue, DWORD *pContrastValue,DWORD *pSaturationValue,DWORD *pHueValue)

参 数： [in]IRealHandle NET_DVR_Login_V40 的返回值
[in]IChannel 通道号
[out] pBrightValue 亮度指针，取值范围[1,10]
[out] pContrastValue 对比度指针，取值范围[1,10]
[out] pSaturationValue 饱和度指针，取值范围[1,10]
[out] pHueValue 色度指针，取值范围[1,10]

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明： 登录设备获取通道的视频参数。

[返回目录](#)

5.3.3 设置预览视频显示参数 **NET_DVR_ClientSetVideoEffect**

函数： BOOL NET_DVR_ClientSetVideoEffect(LONG IRealHandle,DWORD pBrightValue, DWORD pContrastValue,DWORD pSaturationValue,DWORD pHueValue)

参数：

[in]IRealHandle	NET_DVR_RealPlay_V40 的返回值
[in]dwBrightValue	亮度，取值范围[1,10]
[in]dwContrastValue	对比度，取值范围[1,10]
[in]dwSaturationValue	饱和度，取值范围[1,10]
[in]dwHueValue	色度，取值范围[1,10]

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明： 需要预览才能设置视频参数。

[返回目录](#)

5.3.4 设置预览视频显示参数 **NET_DVR_SetVideoEffect**

函数： BOOL NET_DVR_SetVideoEffect(LONG UserID, LONG IChannel,DWORD *pBrightValue, DWORD *pContrastValue,DWORD *pSaturationValue,DWORD *pHueValue)

参数：

[in]IRealHandle	NET_DVR_Login_V40 的返回值
[in]IChannel	通道号
[in]dwBrightValue	亮度，取值范围[1,10]
[in]dwContrastValue	对比度，取值范围[1,10]
[in]dwSaturationValue	饱和度，取值范围[1,10]
[in]dwHueValue	色度，取值范围[1,10]

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明： 登录设备设置通道的视频参数。

[返回目录](#)

5.4 预览画面叠加字符和图像

5.4.1 预览画面叠加字符和图像，Linux 下无此接口 **NET_DVR_RigisterDrawFun**

函数： BOOL NET_DVR_RigisterDrawFun(LONG IRealHandle, fDrawFun cbDrawFun, DWORD dwUser)

参数：

[in]IRealHandle	NET_DVR_RealPlay_V40 的返回值
[in]cbDrawFun	画图回调函数
[in]dwUser	用户数据

```
typedef void(CALLBACK *fDrawFun)(LONG IRealHandle, HDC hDc, DWORD dwUser)
```

[out]IRealHandle	当前的预览句柄
[out]hDc	画图 DC

5.5.4 共享声卡模式下开启声音 **NET_DVR_OpenSoundShare**

函 数: BOOL NET_DVR_OpenSoundShare(LONG IRealHandle)

参 数: [in] IRealHandle NET_DVR_RealPlay_V40 的返回值

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

5.5.5 共享声卡模式下关闭声音 **NET_DVR_CloseSoundShare**

函 数: BOOL NET_DVR_CloseSoundShare (LONG IRealHandle)

参 数: [in] IRealHandle NET_DVR_RealPlay_V40 的返回值

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

5.5.6 调节播放音量 **NET_DVR_Volume**

函 数: BOOL NET_DVR_Volume(LONG IRealHandle,WORD wVolume)

参 数: [in] IRealHandle NET_DVR_RealPlay_V40 的返回值
[in] wVolume 音量, 取值范围[0,0xffff]

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 该接口调节的是 PC 的音量。

[返回目录](#)

5.6 实时预览数据捕获

5.6.1 注册回调函数, 捕获实时码流数据 **NET_DVR_SetRealDataCallBack**

函 数: BOOL NET_DVR_SetRealDataCallBack(LONG IRealHandle, fRealDataCallBack cbRealDataCallBack, DWORD dwUser)

参 数: [in] IRealHandle 预览句柄, NET_DVR_RealPlay_V40 的返回值
[in] cbRealDataCallBack 码流数据回调函数
[in] dwUser 用户数据

```
typedef void(CALLBACK *fRealDataCallBack)(LONG IRealHandle, DWORD dwDataType, BYTE *pBuffer,
DWORD dwBufSize,DWORD dwUser)
```

[out] IRealHandle 当前的预览句柄

[out] dwDataType 数据类型, 详见表 5.2

[out] pBuffer 存放数据的缓冲区指针
 [out] dwBufSize 缓冲区大小
 [out] dwUser 用户数据

表 5.2 码流数据类型

dwDataType 宏定义	宏定义值	含义
NET_DVR_SYSHEAD	1	系统头数据
NET_DVR_STREAMDATA	2	流数据（包括复合流或音视频分开的视频流数据）
NET_DVR_AUDIOSTREAMDATA	3	音频数据

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明：

- 此函数包括开始和停止用户处理 SDK 捕获的数据，当回调函数 cbRealDataCallBack 设为非 NULL 值时，表示回调和处理数据；当设为 NULL 时表示停止回调和处理数据。回调的第一个包是 40 个字节的文件头，供后续解码使用，之后回调的是压缩的码流。回调数据最大为 256K 字节。
- cbRealDataCallBack 回调函数中不能执行可能会占用时间较长的接口或操作，不建议调用该 SDK（HCNetSDK.dll）本身的接口。

[返回目录](#)

5.6.2 捕获数据并保存到指定的文件中 **NET_DVR_SaveRealData**

函数： BOOL NET_DVR_SaveRealData(LONG IRealHandle, char *sFileName)

参数： [in] IRealHandle NET_DVR_RealPlay_V40 的返回值
 [in] sFileName 文件路径指针

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明： V5.0.3.2 或以后版本，通过该接口保存录像，文件最大限制为 1024MB，大于 1024M 时，SDK 自动新建文件进行保存，文件开始将 40 字节头自动写入，文件名命名规则为“在接口传入的文件名基础上增加数字标识(例如：*_1.mp4、*_2.mp4)”。

[返回目录](#)

5.6.3 停止数据捕获 **NET_DVR_StopSaveRealData**

函数： BOOL NET_DVR_StopSaveRealData(LONG IRealHandle)

参数： [in] IRealHandle NET_DVR_RealPlay_V40 的返回值

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明：

[返回目录](#)

5.7 预览时抓图

5.7.1 设置抓图模式 **NET_DVR_SetCapturePictureMode**

函 数: BOOL NET_DVR_SetCapturePictureMode(DWORD dwCaptureMode)

参 数: [in]dwCaptureMode 抓图模式

```
enum tagPDC_PARAM_KEY{
    BMP_MODE    = 0,  // BMP 模式
    JPEG_MODE   = 1  // JPEG 模式
}CAPTURE_MODE
```

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 调用该接口设置抓图模式后, NET_DVR_CapturePicture 可抓取相应的图片。

[返回目录](#)

5.7.2 预览时, 单帧数据捕获图片并保存成文件 **NET_DVR_CapturePicture**

函 数: BOOL NET_DVR_CapturePicture(LONG lRealHandle, char *sPicFileName)

参 数: [in] lRealHandle NET_DVR_RealPlay_V40 的返回值
[in] sPicFileName 保存图象的文件路径, 包含文件名。路径长度和操作系统有关, sdk 不做限制, windows 默认路径长度小于等于 256 字节 (包括文件名在内)。

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

- 在调用该接口之前可以调用 [NET_DVR_SetCapturePictureMode](#) 设置抓图模式, 默认为 BMP 模式。如果抓图模式为 **BMP** 模式, 抓取的是 **BMP** 图片, 保存路径后缀应为.bmp; 如果抓图模式为 **JPEG** 模式, 抓取的是 **JPEG** 图片, 保存路径后缀应为.jpg。
- 调用该接口进行抓图, 实际是播放库解码抓图, 要求在调用 [NET_DVR_RealPlay_V40](#) 等接口时传入非空的播放句柄 (播放库解码显示), 否则时接口会返回失败 (调用次序错误)。
- 智能交通摄像机车牌抓拍请参考 [3.2 报警 \(布防\) 模块流程](#)。

[返回目录](#)

5.8 语音对讲

语音对讲(Windows 32 位系统支持)

5.8.1 启动语音对讲 **NET_DVR_StartVoiceCom_V30**

函 数: LONG NET_DVR_StartVoiceCom_V30(LONG lUserID, DWORD dwVoiceChan, BOOL bNeedCBNoEncData, fVoiceDataCallBack cbVoiceDataCallBack, void* pUser)

参 数:	[in]IUserID	NET_DVR_Login_V40 的返回值
	[in]dwVoiceChan	语音通道号, 从 1 开始
	[in]bNeedCBNoEncData	需要回调的语音数据类型: 0- 编码后的语音数据, 1- 编码前的 PCM 原始数据
	[in]cbVoiceDataCallBack	音频数据回调函数
	[in]pUser	用户数据指针
	typedef void(CALLBACK *fVoiceDataCallBack)(LONG IVoiceComHandle, char *pRecvDataBuffer, DWORD dwBufSize, BYTE byAudioFlag, void *pUser)	
	[out]IVoiceComHandle	NET_DVR_StartVoiceCom_V30 的返回值
	[out]pRecvDataBuffer	存放音频数据的缓冲区指针
	[out]dwBufSize	音频数据大小
	[out]byAudioFlag	音频数据类型: 0- 本地采集的数据, 1- 设备发送过来的语音数据
	[out]pUser	用户数据指针

返回值: -1 表示失败, 其他值作为 NET_DVR_StopVoiceCom 等函数的句柄参数。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: Windows 7 操作系统下, 如果不外接音频设备, 该接口将返回失败。在调用开始语音对讲之前可先配置设备的语音对讲音频编码类型, 即可先调用参数配置中的 NET_DVR_COMPRESSION_AUDIO 结构配置。

当前音频为 G722 编码时, 音频数据的采样频率为 16000, 16 位采样且是单通道的。因此, 音频播放格式应如下定义:

```
const int SAMPLES_PER_SECOND = 16000;
const int CHANNEL = 1;
const int BITS_PER_SAMPLE = 16;
WAVEFORMATEX m_wavFormatEx;
m_wavFormatEx.cbSize = sizeof(m_wavFormatEx);
m_wavFormatEx.nBlockAlign = CHANNEL * BITS_PER_SAMPLE / 8;
m_wavFormatEx.nChannels = CHANNEL;
m_wavFormatEx.nSamplesPerSec = SAMPLES_PER_SECOND;
m_wavFormatEx.wBitsPerSample = BITS_PER_SAMPLE;
m_wavFormatEx.nAvgBytesPerSec = SAMPLES_PER_SECOND*m_wavFormatEx.nBlockAlign
```

当前音频为 G711 或者 G726 编码时, 音频数据的采样频率为 8000, 16 位采样且是单通道的。因此, 音频播放格式应如下定义:

```
const int SAMPLES_PER_SECOND_G711_MU = 8000;
const int CHANNEL = 1;
const int BITS_PER_SAMPLE = 16;
WAVEFORMATEX m_wavFormatEx;
m_wavFormatEx.cbSize = sizeof(m_wavFormatEx);
m_wavFormatEx.nBlockAlign = CHANNEL * BITS_PER_SAMPLE / 8;
m_wavFormatEx.nChannels = CHANNEL;
m_wavFormatEx.nSamplesPerSec = SAMPLES_PER_SECOND_G711_MU;
m_wavFormatEx.wBitsPerSample = BITS_PER_SAMPLE;
m_wavFormatEx.nAvgBytesPerSec = SAMPLES_PER_SECOND_G711_MU*
```

m_wavFormatEx.nBlockAlign;

[返回目录](#)

5.8.2 设置语音对讲客户端的音量 **NET_DVR_SetVoiceComClientVolume**

函 数: BOOL NET_DVR_SetVoiceComClientVolume(LONG lVoiceComHandle, WORD wVolume)
参 数: [in]lVoiceComHandle NET_DVR_StartVoiceCom_V30 的返回值
[in]wVolume 设置音量, 取值范围[0,0xffff]
返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

5.8.3 停止语音对讲或者语音转发 **NET_DVR_StopVoiceCom**

函 数: BOOL NET_DVR_StopVoiceCom(LONG lVoiceComHandle)
参 数: [in]lVoiceComHandle NET_DVR_StartVoiceCom_V30 或
NET_DVR_StartVoiceCom_MR_V30 的返回值
返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

[返回目录](#)

音频压缩参数

5.8.4 获取当前生效的对讲音频压缩参数 **NET_DVR_GetCurrentAudioCompress**

函 数: BOOL NET_DVR_GetCurrentAudioCompress(LONG lUserID, LPNET_DVR_COMPRESSION_AUDIO lpCompressAudio)
参 数: [in] lUserID 用户 ID, NET_DVR_Login_V40 的返回值
[in] lpCompressAudio 音频压缩参数
返回值: -1 表示失败, 其他值为音频编码句柄。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

5.8.5 获取通道参数 **NET_DVR_GetDVRConfig**

函 数: BOOL NET_DVR_GetDVRConfig(LONG lUserID, DWORD dwCommand, LONG lChannel, LPVOID lpOutBuffer, DWORD dwOutBufferSize, LPDWORD lpBytesReturned)
参 数: [in]lUserID 用户 ID 号, NET_DVR_Login_V40 的返回值

[in]dwCommand	设备配置命令，参见配置命令
[in]lChannel	通道号，如果命令不需要通道号，该参数无效，置为 0xFFFFFFFF 即可
[out]lpOutBuffer	接收数据的缓冲指针
[in]dwOutBufferSize	接收数据的缓冲长度(以字节为单位)，不能为 0
[out]lpBytesReturned	实际收到的数据长度指针，不能为 NULL

返回值: TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明: 不同的获取功能对应不同的结构体和命令号，如表 5.3 所示。

表 5.3 获取设备参数

dwCommand 宏定义	dwCommand 含义	通道号	lpOutBuffer 对应结构体	宏定义值
NET_DVR_GET_COMPRESSCFG_AUD	获取语音对讲音频参数	无效	NET_DVR_COMPRESSION_AUDIO	1058

[返回目录](#)

5.9 异常消息回调

5.9.1 注册接收异常、重连等消息回调函数 **NET_DVR_SetExceptionCallBack_V30**

函 数: **Windows 系统下:**

BOOL NET_DVR_SetExceptionCallBack_V30 (UINT nMessage, HWND hWnd, fExceptionCallBack cbExceptionCallBack, void* pUser)

Linux 系统下:

BOOL NET_DVR_SetExceptionCallBack_V30(UINT nMessage, void* hWnd, fExceptionCallBack cbExceptionCallBack, void* pUser)

参 数: [in]nMessage 消息，Linux 下该参数保留
[in]hWnd 接收异常消息的窗口句柄，Linux 下该参数保留
[in]cbExceptionCallBack 接收异常消息的回调函数，回调当前异常的相关信息
[in]pUser 用户数据

typedef void(CALLBACK* fExceptionCallBack)(DWORD dwType, LONG lUserID, LONG lHandle, void* pUser)

[out]dwType 异常或重连等消息的类型，详见表 5.4
[out]lUserID 登录 ID
[out]lHandle 出现异常的相应类型的句柄
[out]pUser 用户数据

表 5.4 异常消息类型

dwType 宏定义	宏定义值	含义
EXCEPTION_EXCHANGE	0x8000	用户交互时异常（注册心跳超时，心跳间隔为 2 分钟）
EXCEPTION_AUDIOEXCHANGE	0x8001	语音对讲异常
EXCEPTION_ALARM	0x8002	报警异常
EXCEPTION_PREVIEW	0x8003	网络预览异常

EXCEPTION_SERIAL	0x8004	透明通道异常
EXCEPTION_RECONNECT	0x8005	预览时重连
EXCEPTION_ALARMRECONNECT	0x8006	报警时重连
EXCEPTION_SERIALRECONNECT	0x8007	透明通道重连
SERIAL_RECONNECTSUCCESS	0x8008	透明通道重连成功
EXCEPTION_PLAYBACK	0x8010	回放异常
EXCEPTION_DISKFORMAT	0x8011	硬盘格式化
EXCEPTION_PASSIVEDECODE	0x8012	被动解码异常
PREVIEW_RECONNECTSUCCESS	0x8015	预览时重连成功
ALARM_RECONNECTSUCCESS	0x8016	报警时重连成功
RESUME_EXCHANGE	0x8017	用户交互恢复
EXCEPTION_MAX_ALARM_INFO	0x8022	报警信息缓存已达上限
EXCEPTION_LOST_ALARM	0x8023	报警丢失

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: Windows 下该函数的 hWnd 和 cbExceptionCallBack 不能同时为 NULL, Linux 下 cbExceptionCallBack 不能设置为 NULL, 否则将接收不到异常消息。

如果此结构是以回调方式反馈异常消息, 那么应用程序中的异常回调函数实现如下, 该函数中的参数 dwType 表示异常消息类型 (见表 5.4); IHandle 表示发生异常的相应类型的句柄。

示例代码:

```
//注册接收异常消息的回调函数
NET_DVR_SetExceptionCallBack_V30(WM_NULL, NULL, g_ExceptionCallBack, NULL);

//接收异常消息的回调函数的外部实现
void CALLBACK g_ExceptionCallBack(DWORD dwType, LONG IUserID, LONG IHandle, void *pUser)
{
    char tempbuf[256];
    ZeroMemory(tempbuf, 256);

    switch(dwType)
    {
        case EXCEPTION_EXCHANGE:
            sprintf(tempbuf, "用户交互时异常!!!");
            TRACE("%s", tempbuf);
            break;

        case EXCEPTION_ALARM:
            sprintf(tempbuf, "报警时网络异常!!!");
            TRACE("%s", tempbuf);
            //TODO: 关闭报警上传
            break;

        case EXCEPTION_ALARMRECONNECT:
```

```

        sprintf(tempbuf,"报警重连!!!");
        break;
    case ALARM_RECONNECTSUCCESS:
        sprintf(tempbuf,"报警重连成功!!!");
        break;
    default:
        break;
    }
}

```

[返回目录](#)

5.10 SDK 日志和信息

5.10.1 获取 SDK 的版本号和 build 信息 **NET_DVR_GetSDKBuildVersion**

函 数： DWORD NET_DVR_GetSDKBuildVersion()

参 数：

返回值： 获取 SDK 的版本号和 build 信息。

说 明： SDK 的版本号和 build 信息。2 个高字节表示版本号：25~32 位表示主版本号，17~24 位表示次版本号；2 个低字节表示 build 信息。如 0x03000101：表示版本号为 3.0，build 号是 0101。

[返回目录](#)

5.10.2 获取当前 SDK 的状态信息 **NET_DVR_GetSDKState**

函 数： BOOL NET_DVR_GetSDKState(LPNET_DVR_SDKSTATE pSDKState);

参 数： [out] pSDKState 状态信息结构，详见：[NET_DVR_SDKSTATE](#)

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说 明：

[返回目录](#)

5.10.3 获取当前 SDK 的功能信息 **NET_DVR_GetSDKAbility**

函 数： BOOL NET_DVR_GetSDKAbility([LPNET_DVR_SDKABL](#) pSDKAbI)

参 数： [out] pSDKAbI 功能信息，详见：[NET_DVR_SDKABL](#)

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说 明：

[返回目录](#)

5.10.4 获取 SDK 本地参数 **NET_DVR_GetSDKLocalCfg**

函 数: `BOOL NET_DVR_GetSDKLocalCfg(NET_SDK_LOCAL_CFG_TYPE enumType, void *lpOutBuff)`
 参 数: `[in] enumType` 配置类型, 不同的取值对应不同的 SDK 参数, 详见表 5.5
`[out] lpOutBuff` 输出参数, 不同的配置类型, 输出参数对应不同的结构, 详见表 5.5
 返回值: `TRUE` 表示成功, `FALSE` 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

表 5.5 本地参数类型

enumType 宏定义	类型值	含义	lpOutBuff 对应结构体
NET_SDK_LOCAL_CFG_TYPE_TCP_PORT_BIND	0	本地 TCP 端口绑定配置	NET_DVR_LOCAL_TCP_PORT_BIND_CFG
NET_SDK_LOCAL_CFG_TYPE_UDP_PORT_BIND	1	本地 UDP 端口绑定配置	NET_DVR_LOCAL_UDP_PORT_BIND_CFG
NET_SDK_LOCAL_CFG_TYPE_MEM_POOL	2	内存池本地配置	NET_DVR_LOCAL_MEM_POOL_CFG
NET_SDK_LOCAL_CFG_TYPE_MODULE_RECV_TIMEOUT	3	按模块配置超时时间	NET_DVR_LOCAL_MODULE_RECV_TIMEOUT_CFG
NET_SDK_LOCAL_CFG_TYPE_ABILITY_PARSE	4	是否使用能力集解析库	NET_DVR_LOCAL_ABILITY_PARSE_CFG
NET_SDK_LOCAL_CFG_TYPE_TALK_MODE	5	对讲模式配置	NET_DVR_LOCAL_TALK_MODE_CFG
NET_SDK_LOCAL_CFG_TYPE_CHECK_DEV	10	心跳交互间隔时间配置	NET_DVR_LOCAL_CHECK_DEV

[返回目录](#)

5.10.5 设置 SDK 本地参数 **NET_DVR_SetSDKLocalCfg**

函 数: `BOOL NET_DVR_SetSDKLocalCfg(NET_SDK_LOCAL_CFG_TYPE enumType, void* const lpInBuff)`
 参 数: `[in] enumType` 配置类型, 不同的取值对应不同的 SDK 参数, 详见表 5.6
`[in] lpInBuff` 输入参数, 不同的配置类型, 输出参数对应不同的结构, 详见表 5.6
 返回值: `TRUE` 表示成功, `FALSE` 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

表 5.6 本地参数类型

enumType 宏定义	类型值	含义	lpInBuff 对应结构体
NET_SDK_LOCAL_CFG_TYPE_TCP_PORT_BIND	0	本地 TCP 端口绑定配置	NET_DVR_LOCAL_TCP_PORT_BIND_CFG
NET_SDK_LOCAL_CFG_TYPE_UDP_PORT_BIND	1	本地 UDP 端口绑定配置	NET_DVR_LOCAL_UDP_PORT_BIND_CFG
NET_SDK_LOCAL_CFG_TYPE_MEM_POOL	2	内存池本地配置	NET_DVR_LOCAL_MEM_POOL_CFG
NET_SDK_LOCAL_CFG_TYPE_MODULE_RECV_TIMEOUT	3	按模块配置超时时间	NET_DVR_LOCAL_MODULE_RECV_TIMEOUT_CFG
NET_SDK_LOCAL_CFG_TYPE_ABILITY_PARSE	4	是否使用能力集解析库	NET_DVR_LOCAL_ABILITY_PARSE_CFG
NET_SDK_LOCAL_CFG_TYPE_TALK_MODE	5	对讲模式配置	NET_DVR_LOCAL_TALK_MODE_CFG
NET_SDK_LOCAL_CFG_TYPE_CHECK_DEV	10	心跳交互间隔时间配置	NET_DVR_LOCAL_CHECK_DEV

NET_SDK_LOCAL_CFG_TYPE_CHAR_ENCODE	13	配置字符编码相关处理回调	NET_DVR_LOCAL_BYTE_ENCODE_CONVERT
------------------------------------	----	--------------	---

[返回目录](#)

5.10.6 启用写日志文件 **NET_DVR_SetLogToFile**

函 数: `BOOL NET_DVR_SetLogToFile(DWORD bLogEnable, char* strLogDir, BOOL bAutoDel)`

参 数: `[in]bLogEnable` 日志的等级（默认为 0）：
 0-表示关闭日志
 1-表示只输出 **ERROR** 错误日志
 2-输出 **ERROR** 错误信息和 **DEBUG** 调试信息
 3-输出 **ERROR** 错误信息、**DEBUG** 调试信息和 **INFO** 普通信息等所有信息
`[in]strLogDir` 日志文件的路径，windows 默认值为"C:\\SdkLog\\"；linux 默认值"/home/sdklog/"
`[in]bAutoDel` 是否删除超出的文件数，默认值为 **TRUE**

返回值: **TRUE** 表示成功，**FALSE** 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说 明: 日志文件路径必须是绝对路径，且以"\"结尾，例如"C:\\SdkLog\\"，建议用户先手动创建文件。若未指定文件路径，则采用默认路径"C:\\SdkLog\\"。可多次调用该接口创建新的日志文件，同时最多支持创建 10 个文件，当设置了删除超出的文件时（即 **bAutoDel** 为 **TRUE**），那么将会自动删除超出的文件。更改目录时到下一次写文件时才会使用新的目录写文件。

[返回目录](#)

5.11 获取错误信息

5.11.1 返回最后操作的错误码 **NET_DVR_GetLastError**

函 数: `DWORD NET_DVR_GetLastError()`

参 数:

返回值: 返回最后操作的错误码。详见[错误码宏定义](#)

说 明: 返回值为错误码，错误码主要分为网络通讯库错误码、RTSP 通讯库错误码和软硬解库错误码。通过错误码可以获取错误信息，然后判断出错原因。

[返回目录](#)

5.11.2 返回最后操作的错误码信息 **NET_DVR_GetErrorMsg**

函 数: `char* NET_DVR_GetErrorMsg(LONG *pErrorNo)`

参 数: `[out] pErrorNo` 错误码数值的指针

返回值: 返回值为错误码信息的指针。错误码主要分为网络通讯库错误码、RTSP 通讯库错误码和软硬解库错误码。详见[错误码宏定义](#)

说明： 通过接口 NET_DVR_GetErrorMsg 直接返回错误信息，直接通过错误信息判断错误原因。

[返回目录](#)

5.12 用户注册

5.12.1 用户注册设备 **NET_DVR_Login_V30**

函数： LONG NET_DVR_Login_V30(char *sDVRIP,WORD wDVRPort, char *sUserName, char *sPassword,[LPNET_DVR_DEVICEINFO_V30](#) lpDeviceInfo)

参数：

[in] Sdvrrip	设备 IP 地址
[in] wDVRPort	设备端口号
[in] sUserName	登录的用户名
[in] sPassword	用户密码
[out] lpDeviceInfo	设备信息，详见： NET_DVR_DEVICEINFO_V30

返回值： -1 表示失败，其他值表示返回的用户 ID 值。该用户 ID 具有唯一性，后续对设备的操作都需要通过此 ID 实现。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明： SDK 注册设备新增支持静态域名的方式，即可设置 sDVRIP="test.vicp.net"。
一台设备同时最多允许 128 个用户注册。
SDK 最多支持 2048 个注册，返回 UserID 的取值范围为 0~2047。

[返回目录](#)

5.12.2 用户注销 **NET_DVR_Logout**

函数： BOOL NET_DVR_Logout(LONG lUserID)

参数： [in] lUserID 用户 ID 号，[NET_DVR_Login_V30](#) 的返回值

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明： 建议使用此接口实现注销功能。

[返回目录](#)

5.13 设备能力集

5.13.1 获取设备能力集 **NET_DVR_GetDeviceAbility**

函数： BOOL NET_DVR_GetDeviceAbility(LONG lUserID, DWORD dwAbilityType, char* pInBuf, DWORD dwInLength, char* pOutBuf, DWORD dwOutLength)

参 数: [in] IUserID 登录主控板, [NET DVR Login V30](#) 的返回值
[in] dwAbilityType 能力类型, 具体定义见表 5.7
[in] pInBuf 输入缓冲区指针, 参见表 5.8
[in] dwInLength 输入缓冲区的长度
[out] pOutBuf 输出缓冲区指针, 参见表 5.8
[in] dwOutLength 接收数据的缓冲区的长度

表 5.7 设备能力集类型

dwAbilityType 宏定义	宏定义值	含义
DEVICE_ALL_ABILITY	0x000	设备全部能力 (保留)
DEVICE_SOFTWARE_ABILITY	0x001	设备软硬件能力
DEVICE_ENCODE_ALL_ABILITY_V20	0x008	设备所有编码能力
DEVICE_USER_ABILITY	0x00c	用户管理参数能力
DEVICE_NETAPP_ABILITY	0x00d	网络应用能力
STREAM_ABILITY	0x012	流能力
DEVICE_SERIAL_ABILITY	0x010	串口能力
SearchLogAbility(DEVICE_ABILITY_IN FO)	0x011	日志搜索能力
ChannelInputAbility(DEVICE_ABILITY _INFO)	0x011	通道输入能力

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET DVR GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 获取各种能力类型时, 输入参数 pInBuf 和输出参数 pOutBuf 的格式定义, 如表 5.8 所示。

表 5.8 设备能力集

能力类型宏定义	能力类型说明	pInBuf	pOutBuf
DEVICE_SOFTWARE_ABILITY	获取设备软硬件能力	无	设备软硬件能力 XML 描述 (BasicCapability)
DEVICE_ENCODE_ALL_ABILITY_V20	设备所有编码能力	编码能力获取输入描述	设备所有编码能力 XML 描述 (AudioVideoCompressInfo)
DEVICE_USER_ABILITY	获取设备用户管理参数能力	用户管理参数能力获取输入描述	设备用户管理参数能力 XML 描述 (UserAbility)
DEVICE_NETAPP_ABILITY	获取设备网络应用参数能力	网络应用参数能力获取输入描述	设备网络应用参数能力 XML 描述 (NetAppAbility)
STREAM_ABILITY	获取设备流能力	获取流能力集	流能力 XML 描述 (StreamAbility)
DEVICE_SERIAL_ABILITY	获取设备 RS232 和 RS485 串口能力	串口能力输入描述	设备串口能力 XML 描述 (SerialAbility)
DEVICE_ABILITY_INFO	设备通用能力类型, 具体能力根据发送的能力节点来区分	获取日志搜索能力集	日志搜索能力 XML 描述 (SearchLogAbility)
DEVICE_ABILITY_INFO	设备通用能力类型, 具体能力根据发送的能力节点来区分	获取设备通道输入能力集	设备通道输入能力 XML 描述 (ChannelInputAbility)

IP_VIEW_DEV_ABILITY	获取可视对讲主机能力集	无	IP 可视对讲主机能力 XML 描述(IpViewDevAbility)
ACS_ABILITY	获取门禁能力集	门禁能力集获取输入描述	门禁能力集 XML 描述(AcsAbility)

注：设备软硬件能力等能力集 XML 描述详细内容请参见《设备网络 SDK 使用手册.chm》。

[返回目录](#)

5.14 布防、撤防

设置报警等信息上传的回调函数

5.14.1 注册回调函数，接收设备报警消息 **NET_DVR_SetDVRMessageCallBack_V30**

函数：BOOL NET_DVR_SetDVRMessageCallBack_V30(MSGCallBack fMessageCallBack, void* pUser)

参数：[in] fMessageCallBack 报警信息回调函数

[in] pUser 用户数据

```
typedef void(CALLBACK *MSGCallBack)(LONG ICommand, NET_DVR_ALARMER *pAlarmer, char *pAlarmInfo, DWORD dwBufLen, void *pUser)
```

[out] ICommand 上传的消息类型，详见表 5.9

[out] pAlarmer 报警设备信息，详见：[NET_DVR_ALARMER](#)

[out] pAlarmInfo 报警信息，详见表 5.10

[out] dwBufLen 报警信息缓存大小

[out] pUser 用户数据

表 5.9 报警布防信息类型

ICommand 宏定义	宏定义值	含义
COMM_UPLOAD_VIDEO_INTERCOM_EVENT	0x1132	可视对讲事件记录信息
COMM_ALARM_VIDEO_INTERCOM	0x1133	可视对讲报警信息

返回值：TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明：该接口中回调函数的第一个参数（ICommand）和第三个参数（pAlarmInfo）是密切关联的，其关系见表 5.10。

表 5.10 报警信息结构

消息类型（ICommand）	上传内容	pAlarmInfo 对应的结构体
COMM_UPLOAD_VIDEO_INTERCOM_EVENT	可视对讲事件记录信息	NET_DVR_VIDEO_INTERCOM_EVENT
COMM_ALARM_VIDEO_INTERCOM	可视对讲报警信息	NET_DVR_VIDEO_INTERCOM_ALARM

LCD 服务器在空闲时会每隔一定时间上传各屏幕工作状态信息。

[返回目录](#)

布防撤防

5.14.2 建立报警上传通道，获取报警等信息 **NET_DVR_SetupAlarmChan_V41**

函 数： LONG NET_DVR_SetupAlarmChan_V41(LONG IUserID, [LPNET_DVR_SETUPALARM_PARAM](#) lpSetupParam)

参 数： [in] IUserID NET_DVR_Login_V30 的返回值
 [in] lpSetupParam 报警布防参数，详见：[NET_DVR_SETUPALARM_PARAM](#)

返回值： -1 表示失败，其他值作为 NET_DVR_CloseAlarmChan_V30 函数的句柄参数。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说 明： 启动布防前，需要调用注册回调函数的接口（如 [NET_DVR_SetDVRMessageCallBack_V30](#)）才能获取到上传的报警信息。
 (别墅门口机 V1.1 新增支持)

[返回目录](#)

5.14.3 撤销报警上传通道 **NET_DVR_CloseAlarmChan_V30**

函 数： BOOL NET_DVR_CloseAlarmChan_V30(LONG IAlarmHandle)

参 数： [in] IAlarmHandle NET_DVR_SetupAlarmChan_V41 的返回值

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说 明：

[返回目录](#)

5.15 监听报警

5.15.1 启动监听，接收设备主动上传的报警等信息 **NET_DVR_StartListen_V30**

函 数： LONG NET_DVR_StartListen_V30(char *sLocalIP, WORD wLocalPort, MSGCallBack DataCallback, void* pUserData = NULL)

参 数： [in] sLocalIP PC 机本地 IP 地址，可以置为 NULL
 [in] wLocalPort PC 本地监听端口号。由用户设置，必须和设备端设置的一致
 [in] DataCallback 回调函数，不能为 NULL
 [in] pUserData 用户数据

typedef void(CALLBACK *MSGCallBack)(LONG ICommand, [NET_DVR_ALARMER](#) *pAlarmer, char *pAlarmInfo, DWORD dwBufLen, void *pUser)

[out] ICommand 上传的消息类型，详见表 5.11

[out] pAlarmer 报警设备信息，详见：[NET_DVR_ALARMER](#)

[out] pAlarmInfo 报警信息，详见表 5.12
 [out] dwBufLen 报警信息缓存大小
 [out] pUser 用户数据

表 5.11 报警监听消息类型

ICommand 宏定义	宏定义值	含义
COMM_UPLOAD_VIDEO_INTERCOM_EVENT	0x1132	可视对讲事件记录信息
COMM_ALARM_VIDEO_INTERCOM	0x1133	可视对讲报警信息

返回值: -1 表示失败，其他值作为 NET_DVR_CloseAlarmChan_V30 函数的句柄参数。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说 明:

- SDK 最大能支持 512 路监听。
- 要使 PC 能够收到设备主动发过来的报警等信息，必须将设备的网络配置中的“远程管理主机地址”或者“远程报警主机地址”设置成 PC 机的 IP 地址（与接口中的 sLocalIP 参数一致），“远程管理主机端口号”或者“远程报警主机端口号”设置成 PC 机的监听端口号（与接口中的 wLocalPort 参数一致）。
- **该接口中的回调函数优先级高于其他回调函数，即设置了该接口中的回调函数，其他回调函数将接收不到报警信息。**
- 接口中回调函数的第一个参数（ICommand）和第三个参数（pAlarmInfo）是密切关联的，其关系见表 5.12。

表 5.12 报警信息结构

消息类型（ICommand）	上传内容	pAlarmInfo 对应的结构体
COMM_UPLOAD_VIDEO_INTERCOM_EVENT	可视对讲事件记录信息	NET_DVR_VIDEO_INTERCOM_EVENT
COMM_ALARM_VIDEO_INTERCOM	可视对讲报警信息	NET_DVR_VIDEO_INTERCOM_ALARM

[返回目录](#)

5.15.2 停止监听（支持多线程）NET_DVR_StopListen_V30

函 数: BOOL NET_DVR_StopListen_V30(LONG lListenHandle)

参 数: [in] lListenHandle 监听句柄，NET_DVR_StartListen_V30 的返回值

返回值: TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说 明:

[返回目录](#)

5.16 通用参数配置

5.16.1 获取设备的配置信息 **NET_DVR_GetDVRConfig**

函 数: BOOL NET_DVR_GetDVRConfig(LONG IUserID, DWORD dwCommand, LONG IChannel, LPVOID lpOutBuffer, DWORD dwOutBufferSize, LPDWORD lpBytesReturned)

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V30 的返回值
[in] dwCommand 设备配置命令, 详见表 5.13
[in] IChannel 通道号或者其他序号, 如果命令不需要通道号, 该参数无效, 置为 0xFFFFFFFF 即可
[out] lpOutBuffer 接收数据的缓冲指针, 详见表 5.13
[in] dwOutBufferSize 接收数据的缓冲长度(以字节为单位), 不能为 0
[out] lpBytesReturned 实际收到的数据长度指针, 不能为 NULL

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 不同的获取功能对应不同的结构体和命令号, 如表 5.13 所示。

表 5.13 参数获取命令列表

dwCommand 宏定义	dwCommand 含义	IChannel	lpOutBuffer 对应结构体	宏定义值
NET_DVR_GET_DEVICECFG_V40	获取设备参数	无效	NET_DVR_DEVICECFG_V40	1100
NET_DVR_GET_TIMECFG	获取时间参数	无效	NET_DVR_TIME	118
NET_DVR_GET_USERCFG_V40	获取用户参数	组号, 从 0 开始, 每组 32 个用户	NET_DVR_USER_V40	6187
NET_DVR_GET_NETCFG_V30	获取网络参数	无效	NET_DVR_NETCFG_V30	1000
NET_DVR_GET_NETAPPCFG	获取网络应用(NTP/DDNS)参数	无效	NET_DVR_NETAPPCFG	222
NET_DVR_GET_NTPCFG	获取网络应用参数(NTP)	无效	NET_DVR_NTPPARA	224
NET_DVR_GET_DDNSCFG_V30	获取网络应用(DDNS)参数	无效	NET_DVR_DDNSPARA_V30	1010

Remark

- 用户名密码采用简单加密方式进行加密解密

[返回目录](#)

5.16.2 设置设备的配置信息 **NET_DVR_SetDVRConfig**

函 数: BOOL NET_DVR_SetDVRConfig(LONG IUserID, DWORD dwCommand, LONG IChannel, LPVOID lpInBuffer, DWORD dwInBufferSize)

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V30 的返回值
[in] dwCommand 设备配置命令, 详见表 5.14
[in] IChannel 通道号, 如果命令不需要通道号, 该参数无效, 置为 0xFFFFFFFF 即可

[in] lpInBuffer 输入数据的缓冲指针, 详见表 5.14

[in] dwInBufferSize 输入数据的缓冲长度(以字节为单位)

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 不同的获取功能对应不同的结构体和命令号, 如表 5.14 所示。

表 5.14 参数配置命令列表

dwCommand 宏定义	dwCommand 含义	IChannel	lpInBuffer 对应结构体	宏定义值
NET_DVR_SET_DEVICECFG_V40	设置设备参数	无效	NET_DVR_DEVICECFG_V40	1101
NET_DVR_SET_TIMECFG	设置设备时间参数	无效	NET_DVR_TIME	119
NET_DVR_SET_USERCFG_V40	设置用户参数	组号, 从 0 开始, 每组 32 个用户	NET_DVR_USER_V40	6188
NET_DVR_SET_NETCFG_V30	设置网络参数	无效	NET_DVR_NETCFG_V30	1001
NET_DVR_SET_NETAPPCFG	设置网络应用(NTP/DDNS)参数	无效	NET_DVR_NETAPPCFG	223
NET_DVR_SET_PICCFG	设置图像参数	有效	NET_DVR_PICCFG	105
NET_DVR_SET_COMPRESSCFG	设置压缩参数	有效	NET_DVR_COMPRESSIONCFG_V30	107
NET_DVR_SET_TIMECFG	设置时间参数	无效	NET_DVR_TIME	119
NET_DVR_SET_POWERON_DELAY_CFG	设置延时开机参数	无效	NET_DVR_DELAY_TIME	9087

[返回目录](#)

5.17 可视对讲参数配置

5.17.1 获取设备的配置信息 **NET_DVR_GetDVRConfig**

函 数: BOOL NET_DVR_GetDVRConfig(LONG IUserID, DWORD dwCommand, LONG IChannel, LPVOID lpOutBuffer, DWORD dwOutBufferSize, LPDWORD lpBytesReturned)

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V30 的返回值
[in] dwCommand 设备配置命令, 详见表 5.13
[in] IChannel 通道号或者其他序号, 如果命令不需要通道号, 该参数无效, 置为 0xFFFFFFFF 即可
[out] lpOutBuffer 接收数据的缓冲指针, 详见表 5.13
[in] dwOutBufferSize 接收数据的缓冲长度(以字节为单位), 不能为 0
[out] lpBytesReturned 实际收到的数据长度指针, 不能为 NULL

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 不同的获取功能对应不同的结构体和命令号, 如表 5.13 所示。

表 5.15 参数获取命令列表

dwCommand 宏定义	dwCommand 含义	IChannel	lpInBuffer 对应结构体	宏定义值
---------------	--------------	----------	------------------	------

NET_DVR_GET_VIDEO_INTERCOM_DEVICEID_CFG	获取可视对讲设备编号	无效	NET_DVR_VIDEO_INTERCOM_DEVICEID_CFG	16001
NET_DVR_GET_OPERATION_TIME_CFG	获取操作时间配置	无效	NET_DVR_VIDEO_INTERCOM_OPERATION_TIME_CFG	16004
NET_DVR_GET_VIDEO_INTERCOM_RELATEDDEV_CFG	获取关联网络设备参数	无效	NET_DVR_VIDEO_INTERCOM_RELATEDDEV_CFG	16006
NET_DVR_GET_VIDEOINTERCOM_STREAM	获取可视对讲流通道参数	通道号	NET_DVR_VIDEOINTERCOM_STREAM	16022
NET_DVR_GET_VIDEO_INTERCOM_IOIN_CFG	获取可视对讲 IO 输入参数	IO 输入号, 从 1 开始	NET_DVR_VIDEO_INTERCOM_IOIN_CFG	16016
NET_DVR_GET_VIDEO_INTERCOM_IOOUT_CFG	获取可视对讲 IO 输出参数	IO 输出号, 从 1 开始	NET_DVR_VIDEO_INTERCOM_IOOUT_CFG	16018
NET_DVR_GET_WDR_CFG	获取宽动态参数	通道号	NET_DVR_WDR_CFG	16024
NET_DVR_GET_ELEVATORCONTROL_CFG_V40	获取梯控器参数	电梯号, 从 1 开始	NET_DVR_ELEVATORCONTROL_CFG_V40	16028
NET_DVR_GET_CALL_ROOM_CFG	获取按键呼叫住户配置	按键编号, 从 1 开始	NET_DVR_CALL_ROOM_CFG	16030
NET_DVR_GET_AUDIOIN_VOLUME_CFG	获取音频输入音量调节参数	音频输入号, 从 1 开始	NET_DVR_VOLUME_CFG	6355
NET_DVR_GET_AUDIOOUT_VOLUME_CFG	获取音频输出音量调节参数	音频输出号, 从 1 开始	NET_DVR_VOLUME_CFG	6369
NET_DVR_GET_VIDEO_INTERCOM_ALARM_CFG	获取可视对讲报警事件参数	无效	NET_DVR_VIDEO_INTERCOM_ALARM_CFG	16037

[返回目录](#)

5.17.2 设置设备的配置信息 **NET_DVR_SetDVRConfig**

函 数: `BOOL NET_DVR_SetDVRConfig(LONG lUserID, DWORD dwCommand, LONG lChannel, LPVOID lpInBuffer, DWORD dwInBufferSize)`

参 数: [in] lUserID 登录主控板, NET_DVR_Login_V30 的返回值
[in] dwCommand 设备配置命令, 详见表 5.14
[in] lChannel 通道号, 如果命令不需要通道号, 该参数无效, 置为 0xFFFFFFFF 即可
[in] lpInBuffer 输入数据的缓冲指针, 详见表 5.14
[in] dwInBufferSize 输入数据的缓冲长度(以字节为单位)

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 不同的获取功能对应不同的结构体和命令号, 如表 5.14 所示。

表 5.16 参数配置命令列表

dwCommand 宏定义	dwCommand 含义	IChannel	IpInBuffer 对应结构体	宏定义值
NET_DVR_SET_VIDEO_INTERCOM_DEVICEID_CFG	设置可视对讲设备编号	无效	NET_DVR_VIDEO_INTERCOM_DEVICEID_CFG	16002
NET_DVR_SET_PRIVILEGE_PASSWORD	设置权限密码配置信息	无效	NET_DVR_PRIVILEGE_PASSWORD_CFG	16003
NET_DVR_SET_OPERATION_TIME_CFG	设置操作时间配置	无效	NET_DVR_VIDEO_INTERCOM_OPERATION_TIME_CFG	16005
NET_DVR_SET_VIDEO_INTERCOM_RELATEDDEV_CFG	设置关联网络设备参数	无效	NET_DVR_VIDEO_INTERCOM_RELATEDDEV_CFG	16007
NET_DVR_SET_VIDEOINTERCOM_STREAM	设置可视对讲流通道参数	通道号, 从 1 开始	NET_DVR_VIDEOINTERCOM_STREAM	16023
NET_DVR_SET_VIDEO_INTERCOM_IOIN_CFG	设置可视对讲 IO 输入参数	IO 输入号, 从 1 开始	NET_DVR_VIDEO_INTERCOM_IOIN_CFG	16017
NET_DVR_SET_VIDEO_INTERCOM_IOOUT_CFG	设置 IO 输出参数	IO 输出号, 从 1 开始	NET_DVR_VIDEO_INTERCOM_IOOUT_CFG	16019
NET_DVR_SET_WDR_CFG	设置宽动态参数	通道号	NET_DVR_WDR_CFG	16025
NET_DVR_SET_ELEVATORCONTROL_CFG_V40	设置梯控器参数	电梯号, 从 1 开始	NET_DVR_ELEVATORCONTROL_CFG_V40	16029
NET_DVR_SET_CALL_ROOM_CFG	设置按键呼叫住户配置	按键编号, 从 1 开始	NET_DVR_CALL_ROOM_CFG	16031
NET_DVR_SET_AUDIOIN_VOLUME_CFG	设置音频输入音量调节参数	音频输入口号, 从 1 开始	NET_DVR_VOLUME_CFG	6356
NET_DVR_SET_AUDIOOUT_VOLUME_CFG	设置音频输出音量调节参数	音频输出口号, 从 1 开始	NET_DVR_VOLUME_CFG	6370
NET_DVR_SET_VIDEO_INTERCOM_ALARM_CFG	设置可视对讲报警事件参数	无效	NET_DVR_VIDEO_INTERCOM_ALARM_CFG	16038

[返回目录](#)

5.18 批量参数配置

5.18.1 批量参数获取(获取 FTP 信息、获取流来源信息)NET_DVR_GetDeviceConfig

函 数: BOOL NET_DVR_GetDeviceConfig(LONG lUserID, DWORD dwCommand, DWORD dwCount, LPVOID lpInBuffer, DWORD dwInBufferSize, LPVOID lpStatusList, LPVOID lpOutBuffer, DWORD dwOutBufferSize)

参 数: [in] lUserID 用户 ID 号, NET_DVR_Login_V30 的返回值

[in] dwCommand	设备配置命令，详见表 5.12
[in] dwCount	批量获取个数，0 和 1 都表示 1 个，依次递增，最大 64 个
[in] lpInBuffer	配置条件缓冲区，详见表 5.13
[in] dwInBufferSize	缓冲区长度
[out] lpStatusList	错误信息列表，和要查询的监控点一一对应，例如 lpStatusList[2] 就对应 lpInBuffer[2]，由用户分配内存，每个错误信息为 4 个字节，参数值：0- 成功，大于 0-失败
[out] lpOutBuffer	设备返回的参数内容（详见表 5.20），和要查询的监控点一一对应。如果某个监控点对应的 lpStatusList 信息为大于 0 值，对应 lpOutBuffer 的内容就是无效的
[in] dwOutBufferSize	输出缓冲区大小

表 5.12 批量参数获取命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_GET_FTPCFG_V40	获取 FTP 信息	6162
NET_DVR_GET_STREAM_SRC_INFO	获取流的来源信息	6017
NET_DVR_GET_ALLSTREAM_SRC_INFO	获取所有流信息（全部获取）	9105

返回值： TRUE 表示成功，但不代表每一个配置都成功，哪一个成功，对应查看 lpStatusList[n]值； FALSE 表示全部失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明： 该接口是带有发送数据的批量获取配置信息的通用接口。lpInBuffer 指定需要获取的 dwCount 个配置信息，lpOutBuffer 保存获取得到的 dwCount 个配置信息。

表 5.13 批量参数获取结构

dwCommand 宏定义	lpInBuffer 对应结构体	lpOutBuffer 对应结构体
NET_DVR_GET_FTPCFG_V40	dwCount 个 NET_DVR_FTP_TYPE	dwCount 个 NET_DVR_FTPCFG_V40
NET_DVR_GET_STREAM_SRC_INFO	dwCount 个 NET_DVR_STREAM_INFO	dwCount 个 NET_DVR_STREAM_SRC_INFO
NET_DVR_GET_ALLSTREAM_SRC_INFO	NULL	4 字节个数(N) + N 个 NET_DVR_STREAM_SRC_CFG

Remark:

- 全部获取时 dwCount 置为 0xffffffff，lpInBuffer 置为 NULL，dwInBufferSize 置为 0，lpStatusList 置为 NULL；lpOutBuffer 前面 4 个字节为个数(N)，后面为设备返回的 N 个信息内容。

[返回目录](#)

5.18.2 批量设置参数(设置 FTP 信息、设置流来源信息)[NET_DVR_SetDeviceConfig](#)

函数： BOOL NET_DVR_SetDeviceConfig(LONG lUserID,DWORD dwCommand,DWORD dwCount,LPOVOID

plnBuffer,DWORD dwInBufferSize,LPVOID lpStatusList,LPVOID lpInParamBuffer,DWORD dwInParamBufferSize)

- 参 数:
- [in] lUserID 用户 ID 号, NET_DVR_Login_V30 的返回值
 - [in] dwCommand 设备配置命令, 详见表 5.14。
 - [in] dwCount 批量配置个数, 0 和 1 都表示 1 个, 依次递增, 最大 64 个
 - [in] plnBuffer 配置条件缓冲区, 详见表 5.15
 - [in] dwInBufferSize 配置条件缓冲区长度
 - [out] lpStatusList 错误信息列表, 和要查询的监控点一一对应, 例如 lpStatusList[2] 就对应 lpInBuffer[2], 由用户分配内存, 每个错误信息为 4 个字节 (1 个 32 位无符号整数值), 参数值: 0 或者 1 表示成功, 其他值为失败对应的错误号
 - [in] lpInParamBuffer 需要设置给设备的参数内容, 和要查询的监控点一一对应。如果某个监控点对应的 lpStatusList 信息为大于 1 的值, 表示对应的 lpInBuffer 设置失败, 为 0 或 1 则表示设置成功, 详见表 5.22
 - [in] dwInParamBufferSize 设置内容缓冲区大小

表 5.14 批量配置

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_SET_STREAM_SRC_INFO	设置流的来源信息	6018
NET_DVR_SET_FTPCFG_V40	设置 FTP 信息	6163

返回值: TRUE 表示成功, 但不代表每一个配置都成功, 哪一个成功, 对应查看 lpStatusList[n]值; FALSE 表示全部失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 该接口是带有发送数据的批量设置监控点配置信息的通用接口扩展接口, 支持设置接收数据超时时间。

表 5.15 批量参数配置结构

dwCommand 宏定义	lpInBuffer 对应结构体	lpInParamBuffer 对应结构体
NET_DVR_SET_STREAM_SRC_INFO	dwCount 个 NET_DVR_STREAM_INFO	dwCount 个 NET_DVR_STREAM_SRC_INFO
NET_DVR_SCREEN_INDEX_SET	dwCount 个 NET_DVR_FTP_TYPE	dwCount 个 NET_DVR_FTPCFG_V40

Remark:

- dwCommand 为 NET_DVR_SET_STREAM_SRC_INFO 时, 如果将 IP 地址、URL 都设为空, 则表示删除流 ID (dwCount 为 0xffff ffff 时表示删除所有流 ID)。

[返回目录](#)

5.19 查询门口机注册信息

5.19.1 启动远程配置（获取门口机注册信息、可视话对讲信令处理）

NET_DVR_StartRemoteConfig

函 数: LONG NET_DVR_StartRemoteConfig (LONG IUserID, DWORD dwCommand, LPVOID lpInBuffer, DWORD dwInBufferLen, fRemoteConfigCallback cbStateCallback, LPVOID pUserData)

参 数: [in] IUserID 用户 ID 号, NET_DVR_Login_V30 返回值
 [in] dwCommand 配置命令, 不同的功能对应不同的命令号(dwCommand), lpInBuffer 等参数也对应不同的内容, 详见表 5.16
 [in] lpInBuffer 输入参数, 具体内容跟配置命令相关
 [in] dwInBufferLen 输入缓冲的大小
 [in] cbStateCallback 状态回调函数
 [in] pUserData 用户数据

```
typedef void(CALLBACK * fRemoteConfigCallback)( DWORD dwType, void*lpBuffer, DWORD dwBufLen, void * pUserData)
```

[out] dwType 配置状态, 具体定义如下所示, 详见表 5.17
 [out] lpBuffer 存放数据的缓冲区指针, 具体内容跟 dwType 相关, 详见列表
 [out] dwBufLen 缓冲区大小
 [out] pUserData 用户数据

表 5.17 门口机注册参数说明

dwCommand 宏定义	宏定义值	含义	lpInBuffer	cbStateCallback
NET_DVR_GET_VIS_REGISTER_INFO	16027	获取门口机注册信息	NULL	NULL
NET_DVR_VIDEO_CALL_SIGNAL_PROCESS	16032	可视话对讲信令处理	NET_DVR_VIDEO_CALL_CONFIG	信息数据

表 5.17 回调状态类型

dwType 宏定义	含义	lpBuffer 对应内容
NET_SDK_CALLBACK_TYPE_STATUS	状态值	回调状态值 <pre>typedef enum { NET_SDK_CALLBACK_STATUS_SUCCESS = 1000, //成功 NET_SDK_CALLBACK_STATUS_PROCESSING, //处理中 NET_SDK_CALLBACK_STATUS_FAILED //失败 NET_SDK_CALLBACK_STATUS_NORMAL }NET_SDK_CALLBACK_STATUS_NORMAL;</pre>
NET_SDK_CALLBACK_TYPE_PROGRESS	进度值	lpBuffer 的值表示进度值(DWORD)
NET_SDK_CALLBACK_TYPE_INFO	信息数	lpBuffer 的值表示信息数据, dwCommand 为

DATA	据	NET_DVR_VIDEO_CALL_SIGNAL_PROCESS 时对应结构体 NET_DVR_VIDEO_CALL_PARAM
------	---	--

返回值： -1 表示失败，其他值作为 NET_DVR_GetNextRemoteConfig 等接口的句柄。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说 明： 调用该接口启动长连接远程配置后，还需要调用其他接口获取或者设置相关参数，如下表所示：

表 5.18 接口调用说明

dwCommand 宏定义	含义	后续接口调用
NET_DVR_GET_VIS_REGISTER_INFO	获取门口机注册信息	NET_DVR_GetNextRemoteConfig
NET_DVR_VIDEO_CALL_SIGNAL_PROCESS	可视对讲信令处理	NET_DVR_SendRemoteConfig

[返回目录](#)

5.19.2 逐个获取查找到的信息 **NET_DVR_GetNextRemoteConfig**

函 数： LONG NET_DVR_GetNextRemoteConfig(LONG lHandle, void *lpOutBuff, DWORD dwOutBuffSize)

参 数： [in] lHandle 查找句柄，NET_DVR_StartRemoteConfig 的返回值
[out] lpOutBuff 输出数据缓冲区，与 NET_DVR_StartRemoteConfig 的命令（dwCommand）有关，详见表 5.19
[out] dwOutBuffSize 缓冲区长度

返回值： -1 表示失败，其他值表示当前的获取状态等信息，详见表 5.18。获取错误码调用 [NET_DVR_GetLastError](#)。

说 明： 调用 NET_DVR_StartRemoteConfig 时传入不同的命令号(dwCommand)，lpOutBuff 对应不同的结构体，如表 5.19 所示。在调用该接口获取查找之前，必须先调用 NET_DVR_StartRemoteConfig 得到当前的查找句柄。此接口用于获取一条已查找到的信息，若要获取全部的已查找到的信息，需要循环调用此接口。

表 5.18 长连接参数获取状态

NET_SDK_GET_NEXT_STATUS_SUCCESS	1000	成功读取到数据，处理完本次数据后需要再次调用 NET_DVR_GetNextRemoteConfig 获取下一条数据
NET_SDK_GET_NEXT_STATUS_NEED_WAIT	1001	需等待设备发送数据，继续调用 NET_DVR_GetNextRemoteConfig
NET_SDK_GET_NEXT_STATUS_FINISH	1002	数据全部取完，可调用 NET_DVR_StopRemoteConfig 结束长连接
NET_SDK_GET_NEXT_STATUS_FAILED	1003	出现异常，可调用 NET_DVR_StopRemoteConfig 结束长连接

表 5.19 长连接参数获取

dwCommand 宏定义	宏定义值	控制功能	lpOutBuff
---------------	------	------	-----------

NET_DVR_GET_VIS_REGISTER_INFO	16027	获取门口机注册信息	NET_DVR_VIS_REGISTER_INFO
-------------------------------	-------	-----------	---

在调用该接口获取查找文件之前，必须先调用 NET_DVR_StartRemoteConfig 得到当前的查找句柄。此接口用于获取一条已查找到的信息，若要获取全部的已查找到的信息，需要循环调用此接口。

[返回目录](#)

5.19.3 发送长连接数据（可视话对讲信令处理）NET_DVR_SendRemoteConfig

函 数： BOOL NET_DVR_SendRemoteConfig(LONG IHandle, DWORD dwDataType, char *pSendBuf, DWORD dwBufSize)

参 数： [in] IHandle 句柄，NET_DVR_StartRemoteConfig 的返回值
[in] dwDataType 数据类型，跟长连接接口 NET_DVR_StartRemoteConfig 的命令参数（dwCommand）有关，详见表 5.21 说明
[in] pSendBuf 保存发送数据的缓冲区，与 dwDataType 有关，详见“Remarks”说明
[in] dwBufSize 缓冲区大小

返回值： TRUE 表示成功，FALSE 表示失败。获取错误码调用 [NET_DVR_GetLastError](#)。

说 明： 不同的数据类型(dwDataType)，pSendBuf 对应不同的结构体，如下表所示：

表 5.21 信令处理参数说明

dwCommand 宏定义	宏定义值	含义	dwDataType 值	pSendBuf 对应结构体
NET_DVR_VIDEO_CALL_SIGNALL_PROCESS	16032	可视话对讲信令处理	0	NET_DVR_VIDEO_CALL_PARAM

[返回目录](#)

5.19.4 关闭长连接 NET_DVR_StopRemoteConfig

函 数： BOOL NET_DVR_StopRemoteConfig(LONG IHandle)

参 数： [in] IHandle 句柄，NET_DVR_StartRemoteConfig 的返回值

返回值： TRUE 表示成功，FALSE 表示失败。获取错误码调用 [NET_DVR_GetLastError](#)。

说 明：

[返回目录](#)

5.20 可视对讲远程控制

远程控制

5.20.1 远程控制 **NET_DVR_RemoteControl**

函 数: BOOL NET_DVR_RemoteControl(LONG IUserID, DWORD dwCommand, LPVOID lpInBuffer, DWORD dwInBufferSize)

参 数: [in] IUserID 用户 ID 号, NET_DVR_Login_V40 返回值
 [in] dwCommand 控制命令, 详见表 5.22
 [in] lpInBuffer 输入参数, 跟控制命令相关, 详见表 5.22
 [in] dwInBufferSize 输入参数长度

返回值: TRUE 表示成功, FALSE 表示失败。获取错误码调用 [NET_DVR_GetLastError](#)。

说 明: 不同的控制功能对应不同的命令号, 同时 lpInBuffer 对应不同的结构体, 详见表 5.16。

表 5.22 远程文件回放控制命令

dwCommand 宏定义	宏定义值	控制功能	lpInBuffer 对应结构体
NET_DVR_REMOTECONTROL_NOTICE_DATA	16008	公告数据下发	NET_DVR_NOTICE_DATA
NET_DVR_REMOTECONTROL_GATEWAY	16009	远程开锁	NET_DVR_CONTROL_GATEWAY
NET_DVR_CLEAR_ACS_PARAM	2118	清空门禁主机参数	NET_DVR_ACS_PARAM_TYPE

[返回目录](#)

5.21 报警门禁相关参数配置

5.21.1 获取报警门禁相关参数 **NET_DVR_GetDVRConfig**

函 数: BOOL NET_DVR_GetDVRConfig(LONG IUserID, DWORD dwCommand, LONG IChannel, LPVOID lpOutBuffer, DWORD dwOutBufferSize, LPDWORD lpBytesReturned)

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V30 的返回值
 [in] dwCommand 设备配置命令, 详见表 5.13
 [in] IChannel 通道号或者其他序号, 如果命令不需要通道号, 该参数无效, 置为 0xFFFFFFFF 即可
 [out] lpOutBuffer 接收数据的缓冲指针, 详见表 5.13
 [in] dwOutBufferSize 接收数据的缓冲长度(以字节为单位), 不能为 0
 [out] lpBytesReturned 实际收到的数据长度指针, 不能为 NULL

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 不同的获取功能对应不同的结构体和命令号, 如表 5.13 所示。

表 5.23 参数获取命令列表

dwCommand 宏定义	dwCommand 含义	IChannel	lpInBuffer 对应结构体	宏定义值
NET_DVR_GET_ALARMIN_PARAM	获取防区参数	防区号, 从 0 开始	NET_DVR_ALARMIN_PARAM	1183
NET_DVR_GET_DOOR_CFG	获取门参数	门编号, 从 1 开始	NET_DVR_DOOR_CFG	2108
NET_DVR_GET_SIREN_PARAM	获取警号参数	1 个警号, 设为 0	NET_DVR_SIREN_PARAM	1187
NET_DVR_GET_ALARM_RS485CFG	获取报警主机 RS485 参数	485 通道号, 从 1 开始	NET_DVR_ALARM_RS485CFG	1189

[返回目录](#)

5.21.2 设置报警门禁相关参数 **NET_DVR_SetDVRConfig**

函 数: BOOL NET_DVR_SetDVRConfig(LONG IUserID, DWORD dwCommand, LONG IChannel, LPVOID lpInBuffer, DWORD dwInBufferSize)

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V30 的返回值
[in] dwCommand 设备配置命令, 详见表 5.14
[in] IChannel 通道号, 如果命令不需要通道号, 该参数无效, 置为 0xFFFFFFFF 即可
[in] lpInBuffer 输入数据的缓冲指针, 详见表 5.14
[in] dwInBufferSize 输入数据的缓冲长度(以字节为单位)

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 不同的获取功能对应不同的结构体和命令号, 如表 5.14 所示。

表 5.24 参数配置命令列表

dwCommand 宏定义	dwCommand 含义	IChannel	lpInBuffer 对应结构体	宏定义值
NET_DVR_SET_ALARM_RS485CFG	设置报警主机 RS485 参数	485 通道号, 从 1 开始	NET_DVR_ALARM_RS485CFG	1188
NET_DVR_SET_ALARMIN_PARAM	设置设备时间参数	无效	NET_DVR_TIME	119
NET_DVR_SET_ALARMIN_PARAM	设置防区参数	防区号, 从 0 开始	NET_DVR_ALARMIN_PARAM	1182
NET_DVR_SET_DOOR_CFG	设置门参数	门编号, 从 1 开始	NET_DVR_DOOR_CFG	2109

[返回目录](#)

5.21.3 启动远程配置（获取设置卡参数） **NET_DVR_StartRemoteConfig**

函 数: LONG NET_DVR_StartRemoteConfig (LONG IUserID, DWORD dwCommand, LPVOID lpInBuffer, DWORD dwInBufferLen, fRemoteConfigCallback cbStateCallback, LPVOID pUserData)

参 数: [in] IUserID 用户 ID 号, NET_DVR_Login_V30 返回值
[in] dwCommand 配置命令, 不同的功能对应不同的命令号(dwCommand), lpInBuffer

	等参数也对应不同的内容，详见表 5.25
[in] lpInBuffer	输入参数，具体内容跟配置命令相关
[in] dwInBufferLen	输入缓冲的大小
[in] cbStateCallback	状态回调函数
[in] pUserData	用户数据

```
typedef void(CALLBACK * fRemoteConfigCallback)( DWORD dwType, void*lpBuffer, DWORD dwBufLen, void * pUserData)
```

[out] dwType	配置状态，具体定义如下所示，详见表 5.26
[out] lpBuffer	存放数据的缓冲区指针，具体内容跟 dwType 相关，详见列表
[out] dwBufLen	缓冲区大小
[out] pUserData	用户数据

表 5.25 卡参数说明

dwCommand 宏定义	宏定义值	含义	lpInBuffer	cbStateCallback
NET_DVR_GET_CARD_CFG	2116	获取卡参数	NET_DVR_CARD_CFG_COND	返回状态、信息数据
NET_DVR_SET_CARD_CFG	2117	设置卡参数	NET_DVR_CARD_CFG_COND	返回状态

表 5.26 回调状态类型

dwType 宏定义	含义	lpBuffer 对应内容
NET_SDK_CALLBACK_TYPE_STATUS	状态值	回调状态值 <pre>typedef enum { NET_SDK_CALLBACK_STATUS_SUCCESS = 1000, //表示获取和配置成功并且结束 NET_SDK_CALLBACK_STATUS_PROCESSING, lpBuffer:4 字节状态+32 字节卡号 //处理中 NET_SDK_CALLBACK_STATUS_FAILED, lpBuffer:4 字节状态+4 字节错误码+32 字节卡号 //失败 NET_SDK_CALLBACK_STATUS_EXCEPTION, 表示长连接配置异常 }NET_SDK_CALLBACK_STATUS_NORMAL;</pre>
NET_SDK_CALLBACK_TYPE_PROGRESS	进度值	lpBuffer 的值表示进度值(DWORD)
NET_SDK_CALLBACK_TYPE_DATA	信息数据	lpBuffer 的值表示信息数据， NET_DVR_GET_CARD_CFG 时，对应结构体： NET_DVR_CARD_CFG

返回值：-1 表示失败，其他值作为 NET_DVR_SendRemoteConfig 等接口的句柄。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明：

- NET_DVR_GET_CARD_CFG 获取卡参数时，在调用该接口启动长连接远程配置后，还需要调用 NET_DVR_SendRemoteConfig 发送查找条件数据(获取所有卡参数时不需要调用该发送接口)，查找结果在 NET_DVR_StartRemoteConfig 设置的回调函数中返回。NET_DVR_SET_CARD_CFG 设置卡参数时，在调用该接口启动长连接远程配置后，通过调用 NET_DVR_SendRemoteConfig 向设备下发卡参数

信息。

[返回目录](#)

5.21.4 发送长连接数据（获取设置卡参数）**NET_DVR_SendRemoteConfig**

函 数: BOOL NET_DVR_SendRemoteConfig(LONG IHandle, DWORD dwDataType, char *pSendBuf, DWORD dwBufSize)

参 数: [in] IHandle 句柄, NET_DVR_StartRemoteConfig 的返回值
 [in] dwDataType 数据类型, 跟长连接接口 NET_DVR_StartRemoteConfig 的命令参数 (dwCommand) 有关, 详见表 5.27 说明
 [in] pSendBuf 保存发送数据的缓冲区, 与 dwDataType 有关, 详见“Remarks”说明
 [in] dwBufSize 缓冲区大小

返回值: TRUE 表示成功, FALSE 表示失败。获取错误码调用 [NET_DVR_GetLastError](#)。

说 明: 长连接接口 NET_DVR_StartRemoteConfig 中不同的命令(dwCommand), 对应不同的数据类型(dwDataType)和发送数据内容(pSendBuf 对应不同的结构体), 如下表所示:

表 5.27 参数说明

dwCommand 宏定义	宏定义值	含义	dwDataType 宏定义	宏定义值	含义	pSendBuf 对应结构体
NET_DVR_GET_CARD_CFG	2116	获取卡参数	ENUM_ACS_SEND_DATA	0x3	门禁主机数据类型	NET_DVR_CARD_CFG_SEND_DATA
NET_DVR_SET_CARD_CFG	2117	设置卡参数	ENUM_ACS_SEND_DATA	0x3	门禁主机数据类型	NET_DVR_CARD_CFG

- 在调用该接口之前, 必须先调用 NET_DVR_StartRemoteConfig 获取长连接句柄。
- NET_DVR_GET_CARD_CFG 获取卡参数或者 NET_DVR_GET_CARD_PASSWD_CFG 获取卡密码开门使能参数时, pSendBuf 为查找条件, 查找到的卡参数信息或者卡密码开门使能信息在 NET_DVR_StartRemoteConfig 设置的回调函数中返回。
- NET_DVR_SET_CARD_CFG 设置卡参数时, pSendBuf 为下发的卡参数信息, 必须保证卡号是从小到大递增的(可以不连续)而且卡号的整型值不能重复(比如不能同时含有 1 和 01 两种卡号), 否则将返回失败。

[返回目录](#)

5.21.5 关闭长连接**NET_DVR_StopRemoteConfig**

函 数: BOOL NET_DVR_StopRemoteConfig(LONG IHandle)

参 数: [in] IHandle 句柄, NET_DVR_StartRemoteConfig 的返回值

返回值: TRUE 表示成功, FALSE 表示失败。获取错误码调用 [NET_DVR_GetLastError](#)。

说 明:

[返回目录](#)

[返回目录](#)

5.22 透明通道

5.22.1 建立透明通道 **NET_DVR_SerialStart_V40**

函 数: `BOOL NET_DVR_SerialStart_V40 (LONG IUserID, void *IpInBuffer, LONG dwInBufferSize, fSerialDataCallBack cbSerialDataCallBack, void *pUser)`

参 数:

[in] IUserID	用户 ID, NET_DVR_Login_V30 的返回值
[in] dwCommand	输入缓冲区, 指向透明通道串口参数, 对应 NET_DVR_SERIALSTART_V40
[in] dwInBufferSize	输入缓冲区大小
[in] cbSerialDataCallBack	串口数据回调函数
[in] pUser	用户数据

```
typedef void(CALLBACK *fSerialDataCallBack)(LONG ISerialHandle, LONG IChannel, char *pRecvDataBuffer, DWORD dwBufSize, void *pUser);
```

[out] ISerialHandle	NET_DVR_SerialStart_V40 的返回值
[out] IChannel	通道号, 透明通道类型为 RS485 通道时有效
[out] pRecvDataBuffer	存放数据的缓冲区指针
[out] dwBufSize	数据大小
[out] pUser	用户数据

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 如果串口接的外设不支持双工 (全双工或者半双工), 设备不会有数据返回给 SDK。

[返回目录](#)

5.22.2 通过透明通道向设备串口发送数据 **NET_DVR_SerialSend**

函 数: `BOOL NET_DVR_SerialSend (LONG ISerialHandle, LONG IChannel, char *pSendBuf, DWORD dwBufSize)`

参 数:

[in] ISerialHandle	NET_DVR_SerialStart 的返回值
[in] IChannel	串口号, 从 1 开始
[in] pSendBuf	发送数据的缓冲区指针
[in] dwBufSize	缓冲区的大小, 最多 1016 字节

返回值: TRUE 表示成功, ; FALSE 表示全部失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

5.22.3 断开透明通道 **NET_DVR_SerialStop**

函 数: BOOL NET_DVR_SerialStop (LONG ISerialHandle)

参 数: [in] ISerialHandle NET_DVR_SerialStart 的返回值

返回值: TRUE 表示成功,; FALSE 表示全部失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

5.23 设备维护

远程升级

5.23.1 设置远程升级时网络环境 **NET_DVR_SetNetworkEnvironment**

函 数: BOOL NET_DVR_SetNetworkEnvironment(DWORD dwEnvironmentLevel)

参 数: [in]dwEnvironmentLevel 网络环境级别

```
enum{
    LOCAL_AREA_NETWORK = 0, //局域网环境
    WIDE_AREA_NETWORK    //广域网环境
}
```

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 接口中的网络环境级别参数分为两类:

[LOCAL_AREA_NETWORK](#) 表示局域网环境(网络环境好, 通讯流畅);

[WIDE_AREA_NETWORK](#) 表示广域网环境(网络环境差, 易阻塞)。

在调用远程升级接口之前, 可以通过此接口适应不同的升级环境。

[返回目录](#)

5.23.2 远程升级 **NET_DVR_Upgrade**

函 数: LONG NET_DVR_Upgrade(LONG IUserID, char *sFileName)

参 数: [in]IUserID NET_DVR_Login_V30 的返回值

[in]sFileName 升级的文件路径 (包括文件名)。路径长度和操作系统有关, sdk 不做限制, windows 默认路径长度小于等于 256 字节 (包括文件名在内)。

返回值: -1 表示失败, 其他值作为 NET_DVR_GetUpgradeState 等函数的参数。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

5.23.3 获取远程升级的进度 **NET_DVR_GetUpgradeProgress**

函 数: Int NET_DVR_GetUpgradeProgress(LONG IUpgradeHandle)

参 数: [in]IUpgradeHandle NET_DVR_Upgrade 的返回值

返回值: -1 表示失败, 0~100 表示升级进度。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

5.23.4 获取远程升级的状态 **NET_DVR_GetUpgradeState**

函 数: int NET_DVR_GetUpgradeState(LONG IUpgradeHandle)

参 数: [in]IUpgradeHandle NET_DVR_Upgrade 的返回值

返回值: -1 表示失败, 其他值定义如下: 1- 升级成功; 2- 正在升级; 3- 升级失败; 4- 网络断开, 状态未知; 5- 升级文件语言版本不匹配。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

5.23.5 获取远程升级的阶段信息 **NET_DVR_GetUpgradeStep**

函 数: LONG NET_DVR_GetUpgradeStep(LONG IUpgradeHandle, LONG *pSubProgress)

参 数: [in]IUpgradeHandle NET_DVR_Upgrade 的返回值

[in]pSubProgress 升级阶段子进度

返回值: -1 表示失败, 其他值定义如表 5.28 所示。

表 5.28 升级阶段

宏定义	宏定义值	含义
STEP_RECV_DATA	1	接收升级包数据
STEP_UPGRADE	2	升级系统
STEP_BACKUP	3	备份系统
STEP_SEARCH	255	设备正在搜索升级文件

接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

5.23.6 关闭远程升级句柄，释放资源 **NET_DVR_CloseUpgradeHandle**

函 数： BOOL NET_DVR_CloseUpgradeHandle(LONG lUpgradeHandle)
参 数： [in]lUpgradeHandle NET_DVR_Upgrade 的返回值
返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。
说 明：

[返回目录](#)

日志查找

5.23.7 查找设备的日志信息 **NET_DVR_FindDVRLog_V30**

函 数： LONG NET_DVR_FindDVRLog_V30(LONG lUserID, LONG lSelectMode, DWORD dwMajorType, DWORD dwMinorType, [LPNET_DVR_TIME](#) lpStartTime, [LPNET_DVR_TIME](#) lpStopTime, BOOL bOnlySmart = FALSE)
参 数： [in]lUserID NET_DVR_Login_V30 的返回值
[in]lSelectMode 查询方式：0- 全部，1- 按类型，2- 按时间，3- 按时间和类型
[in]dwMajorType 日志主类型（S.M.A.R.T 搜索时无效），0 表示全部类型，其他类型定义见[日志主类型列表](#)
[in]dwMinorType 日志次类型（S.M.A.R.T 搜索时无效），0 表示全部类型，根据不同的主类型的次类型定义见[日志次类型列表](#)
[in]lpStartTime 文件的开始时间，详见：[NET_DVR_TIME](#)
[in]lpStopTime 文件结束时间，详见：[NET_DVR_TIME](#)
[in]bOnlySmart 是否只搜索带 S.M.A.R.T 信息的日志
返回值： -1 表示失败，其他值作为 NET_DVR_FindNextLog_V30 等函数的参数。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。
说 明： 通常不需要搜索详细的 S.M.A.R.T 信息时，置 bOnlySmart 为 FALSE 即可完成所有日志信息的搜索。S.M.A.R.T 信息：硬盘运行日志记录。

[返回目录](#)

5.23.8 逐条获取查找到的日志信息 **NET_DVR_FindNextLog_V30**

函 数： LONG NET_DVR_FindNextLog_V30(LONG lLogHandle, [LPNET_DVR_LOG_V30](#) lpLogData)
参 数： [in]lLogHandle 日志查找句柄，NET_DVR_FindDVRLog_V30 的返回值
[out]lpLogData 保存日志信息的指针，详见：[NET_DVR_LOG_V30](#)
返回值： -1 表示失败，其他值表示当前的获取状态等信息，详见表 5.29。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

表 5.29 状态信息

宏定义	宏定义值	含义
NET_DVR_FILE_SUCCESS	1000	获取日志信息成功

NET_DVR_FILE_NOFIND	1001	未查找到日志
NET_DVR_ISFINDING	1002	正在查找请等待
NET_DVR_NOMOREFILE	1003	没有更多的日志，查找结束
NET_DVR_FILE_EXCEPTION	1004	查找日志时异常

说明： 在调用该接口获取查找日志之前，必须先调用 [NET_DVR_FindDVRLog_V30](#) 得到当前的查找句柄。查找结束后调用 [NET_DVR_FindLogClose_V30](#) 释放资源。

[返回目录](#)

5.23.9 释放查找日志的资源 **NET_DVR_FindLogClose_V30**

函数： BOOL NET_DVR_FindLogClose_V30(LONG lLogHandle)

参数： [in] lLogHandle 日志查找句柄，NET_DVR_FindDVRLog_V30 的返回值

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明：

[返回目录](#)

远程恢复默认参数、重启

5.23.10 恢复设备默认参数 **NET_DVR_RestoreConfig**

函数： BOOL NET_DVR_RestoreConfig(LONG lUserID)

参数： [in] lUserID 用户 ID 号，NET_DVR_Login_V30 的返回值

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明：

[返回目录](#)

5.23.11 重启设备 **NET_DVR_RebootDVR**

函数： BOOL NET_DVR_RebootDVR(LONG lUserID)

参数： [in] lUserID 用户 ID 号，NET_DVR_Login_V30 的返回值

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明：

[返回目录](#)

配置文件导入导出

5.23.12 导出配置文件 **NET_DVR_GetConfigFile**

函 数: `BOOL NET_DVR_GetConfigFile(LONG lUserID, char *sFileName)`

参 数: `[in]lUserID` 用户 ID 号, `NET_DVR_Login_V30` 的返回值
`[in]sFileName` 存放保存配置文件的文件路径 (二进制文件)

返回值: `TRUE` 表示成功, `FALSE` 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

5.23.13 导入配置文件 **NET_DVR_SetConfigFile**

函 数: `BOOL NET_DVR_SetConfigFile(LONG lUserID, char *sFileName)`

参 数: `[in]lUserID` 用户 ID 号, `NET_DVR_Login_V30` 的返回值
`[in]sFileName` 存放保存配置文件的文件路径 (二进制文件)

返回值: `TRUE` 表示成功, `FALSE` 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

6 错误代码及说明

6.1 网络通讯库错误码

错误名称	错误值	说明
NET_DVR_NOERROR	0	没有错误。
NET_DVR_PASSWORD_ERROR	1	用户名密码错误。注册时输入的用户名或者密码错误。
NET_DVR_NOENOUGHPRI	2	权限不足。该注册用户没有权限执行当前对设备的操作，可以与远程用户参数配置做对比。
NET_DVR_NOINIT	3	SDK 未初始化。
NET_DVR_CHANNEL_ERROR	4	通道号错误。设备没有对应的通道号。
NET_DVR_OVER_MAXLINK	5	连接到设备的用户个数超过最大。
NET_DVR_VERSIONNOMATCH	6	版本不匹配。SDK 和设备的版本不匹配。
NET_DVR_NETWORK_FAIL_CONNECT	7	连接设备失败。设备不在线或网络原因引起的连接超时等。
NET_DVR_NETWORK_SEND_ERROR	8	向设备发送失败。
NET_DVR_NETWORK_RECV_ERROR	9	从设备接收数据失败。
NET_DVR_NETWORK_RECV_TIMEOUT	10	从设备接收数据超时。
NET_DVR_NETWORK_ERRORDATA	11	传送的数据有误。发送给设备或者从设备接收到的数据错误，如远程参数配置时输入设备不支持的值。
NET_DVR_ORDER_ERROR	12	调用次序错误。
NET_DVR_OPERNOPERMIT	13	无此权限。
NET_DVR_COMMANDTIMEOUT	14	设备命令执行超时。
NET_DVR_ERRORSERIALPORT	15	串口号错误。指定的设备串口号不存在。
NET_DVR_ERRORALARMPORT	16	报警端口错误。指定的设备报警输出端口不存在。
NET_DVR_PARAMETER_ERROR	17	参数错误。SDK 接口中给入的输入或输出参数为空。
NET_DVR_CHAN_EXCEPTION	18	设备通道处于错误状态
NET_DVR_NODISK	19	设备无硬盘。当设备无硬盘时，对设备的录像文件、硬盘配置等操作失败。
NET_DVR_ERRORDISKNUM	20	硬盘号错误。当对设备进行硬盘管理操作时，指定的硬盘号不存在时返回该错误。
NET_DVR_DISK_FULL	21	设备硬盘满。
NET_DVR_DISK_ERROR	22	设备硬盘出错
NET_DVR_NOSUPPORT	23	设备不支持。
NET_DVR_BUSY	24	设备忙。
NET_DVR_MODIFY_FAIL	25	设备修改不成功。
NET_DVR_PASSWORD_FORMAT_ERROR	26	密码输入格式不正确
NET_DVR_DISK_FORMATING	27	硬盘正在格式化，不能启动操作。
NET_DVR_DVRNORESOURCE	28	设备资源不足。
NET_DVR_DVROPRATEFAILED	29	设备操作失败。
NET_DVR_OPENHOSTSOUND_FAIL	30	语音对讲、语音广播操作中采集本地音频或打开音频输出失败。
NET_DVR_DVRVOICEOPENED	31	设备语音对讲被占用。

NET_DVR_TIMEINPUTERROR	32	时间输入不正确。
NET_DVR_NOSPECFILE	33	回放时设备没有指定的文件。
NET_DVR_CREATEFILE_ERROR	34	创建文件出错。本地录像、保存图片、获取配置文件和远程下载录像时创建文件失败。
NET_DVR_FILEOPENFAIL	35	打开文件出错。设置配置文件、设备升级、上传审讯文件时打开文件失败。
NET_DVR_OPERNOTFINISH	36	上次的操作还没有完成
NET_DVR_GETPLAYTIMEFAIL	37	获取当前播放的时间出错。
NET_DVR_PLAYFAIL	38	播放出错。
NET_DVR_FILEFORMAT_ERROR	39	文件格式不正确。
NET_DVR_DIR_ERROR	40	路径错误
NET_DVR_ALLOC_RESOURCE_ERROR	41	SDK 资源分配错误。
NET_DVR_AUDIO_MODE_ERROR	42	声卡模式错误。当前打开声音播放模式与实际设置的模式不符出错。
NET_DVR_NOENOUGH_BUF	43	缓冲区太小。接收设备数据的缓冲区或存放图片缓冲区不足。
NET_DVR_CREATESOCKET_ERROR	44	创建 SOCKET 出错。
NET_DVR_SETSOCKET_ERROR	45	设置 SOCKET 出错。
NET_DVR_MAX_NUM	46	个数达到最大。分配的注册连接数、预览连接数超过 SDK 支持的最大数。
NET_DVR_USERNOTEXIST	47	用户不存在。注册的用户 ID 已注销或不可用。
NET_DVR_WRITEFLASHERROR	48	写 FLASH 出错。设备升级时写 FLASH 失败。
NET_DVR_UPGRADEFAIL	49	设备升级失败。网络或升级文件语言不匹配等原因升级失败。
NET_DVR_CARDHAVEINIT	50	解码卡已经初始化过。
NET_DVR_PLAYERFAILED	51	调用播放库中某个函数失败。
NET_DVR_MAX_USERNUM	52	登录设备的用户数达到最大。
NET_DVR_GETLOCALIPANDMACFAIL	53	获得本地 PC 的 IP 地址或物理地址失败。
NET_DVR_NOENCODEING	54	设备该通道没有启动编码。
NET_DVR_IPMISMATCH	55	IP 地址不匹配。
NET_DVR_MACMISMATCH	56	MAC 地址不匹配。
NET_DVR_UPGRADELANGMISMATCH	57	升级文件语言不匹配。
NET_DVR_MAX_PLAYERPORT	58	播放器路数达到最大。
NET_DVR_NOSPACEBACKUP	59	备份设备中没有足够空间进行备份。
NET_DVR_NODEVICEBACKUP	60	没有找到指定的备份设备。
NET_DVR_PICTURE_BITS_ERROR	61	图像像素位数不符，限 24 色。
NET_DVR_PICTURE_DIMENSION_ERROR	62	图片高*宽超限，限 128*256。
NET_DVR_PICTURE_SIZ_ERROR	63	图片大小超限，限 100K。
NET_DVR_LOADPLAYERSDKFAILED	64	载入当前目录下 Player Sdk 出错。
NET_DVR_LOADPLAYERSDKPROC_ERROR	65	找不到 Player Sdk 中某个函数入口。
NET_DVR_LOADDSSDKFAILED	66	载入当前目录下 DsSdk 出错。
NET_DVR_LOADDSSDKPROC_ERROR	67	找不到 DsSdk 中某个函数入口。
NET_DVR_DSSDK_ERROR	68	调用硬解码库 DsSdk 中某个函数失败。

NET_DVR_VOICEMONOPOLIZE	69	声卡被独占。
NET_DVR_JOINMULTICASTFAILED	70	加入多播组失败。
NET_DVR_CREATEDIR_ERROR	71	建立日志文件目录失败。
NET_DVR_BINDSOCKET_ERROR	72	绑定套接字失败。
NET_DVR_SOCKETCLOSE_ERROR	73	socket 连接中断，此错误通常是由于连接中断或目的地不可达。
NET_DVR_USERID_ISUSING	74	注销时用户 ID 正在进行某操作。
NET_DVR_SOCKETLISTEN_ERROR	75	监听失败。
NET_DVR_PROGRAM_EXCEPTION	76	程序异常。
NET_DVR_WRITEFILE_FAILED	77	写文件失败。本地录像、远程下载录像、下载图片等操作时写文件失败。
NET_DVR_FORMAT_READONLY	78	禁止格式化只读硬盘。
NET_DVR_WITHSAMEUSERNAME	79	远程用户配置结构中存在相同的用户名。
NET_DVR_DEVICEYPE_ERROR	80	导入参数时设备型号不匹配。
NET_DVR_LANGUAGE_ERROR	81	导入参数时语言不匹配。
NET_DVR_PARAVERSION_ERROR	82	导入参数时软件版本不匹配。
NET_DVR_IPCHAN_NOTALIVE	83	预览时外接 IP 通道不在线。
NET_DVR_RTSP_SDK_ERROR	84	加载标准协议通讯库 StreamTransClient 失败。
NET_DVR_CONVERT_SDK_ERROR	85	加载转封装库失败。
NET_DVR_IPC_COUNT_OVERFLOW	86	超出最大的 IP 接入通道数。
NET_DVR_MAX_ADD_NUM	87	添加录像标签或者其他操作超出最多支持的个数。
NET_DVR_PARAMMODE_ERROR	88	图像增强仪，参数模式错误（用于硬件设置时，客户端进行软件设置时错误值）。
NET_DVR_CODESPITTER_OFFLINE	89	码分器不在线。
NET_DVR_BACKUP_COPYING	90	设备正在备份。
NET_DVR_CHAN_NOTSUPPORT	91	通道不支持该操作。
NET_DVR_CALLINEINVALID	92	高度线位置太集中或长度线不够倾斜。
NET_DVR_CALCANCELCONFLICT	93	取消标定冲突，如果设置了规则及全局的实际大小尺寸过滤。
NET_DVR_CALPOINTOUTRANGE	94	标定点超出范围。
NET_DVR_FILTERRECTINVALID	95	尺寸过滤器不符合要求。
NET_DVR_DDNS_DEVOFFLINE	96	设备没有注册到 ddns 上。
NET_DVR_DDNS_INTER_ERROR	97	DDNS 服务器内部错误。
NET_DVR_ALIAS_DUPLICATE	150	别名重复（EasyDDNS 的配置）
NET_ERROR_TRUNK_LINE	711	子系统已被配成干线
NET_ERROR_MIXED_JOINT	712	不能进行混合拼接
NET_ERROR_DISPLAY_SWITCH	713	不能进行显示通道切换
NET_ERROR_USED_BY_BIG_SCREEN	714	解码资源被大屏占用
NET_ERROR_USE_OTHER_DEC_RESOURCE	715	不能使用其他解码子系统资源
NET_ERROR_DISP_MODE_SWITCH	716	显示通道显示状态切换中
NET_ERROR_SCENE_USING	717	场景正在使用
NET_ERR_NO_ENOUGH_DEC_RESOURCE	718	解码资源不足
NET_ERR_NO_ENOUGH_FREE_SHOW_RESOURCE	719	畅显资源不足
NET_ERR_NO_ENOUGH_VIDEO_MEMORY	720	显存资源不足

NET_ERR_MAX_VIDEO_NUM	721	一拖多资源不足
NET_ERR_WIN_COVER_FREE_SHOW_AND_NORMAL	722	窗口跨越了畅显输出口和非畅显输出口
NET_ERR_FREE_SHOW_WIN_SPLIT	723	畅显窗口不支持分屏
NET_ERR_INAPPROPRIATE_WIN_FREE_SHOW	724	不是输出口整数倍的窗口不支持开启畅显
NET_DVR_TRANSPARENT_WIN_NOT_SUPPORT_SPLIT	725	开启透明度的窗口不支持分屏
NET_DVR_SPLIT_WIN_NOT_SUPPORT_TRANSPARENT	726	开启多分屏的窗口不支持透明度设置
NET_ERR_MAX_LOGO_NUM	727	Logo 数达到上限
NET_ERR_MAX_WIN_LOOP_NUM	728	轮巡窗口数达到上限
NET_DVR_DEV_NET_OVERFLOW	800	网络流量超过设备能力上限
NET_DVR_STATUS_RECORDFILE_WRITING_NOT_LOCK	801	录像文件在录像，无法被锁定
NET_DVR_STATUS_CANT_FORMAT_LITTLE_DISK	802	由于硬盘太小无法格式化
NET_ERR_WINCHAN_IDX	901	开窗通道号错误
NET_ERR_WIN_LAYER	902	窗口层数错误，单个屏幕上最多覆盖的窗口层数
NET_ERR_WIN_BLK_NUM	903	窗口的块数错误，单个窗口可覆盖的屏幕个数
NET_ERR_OUTPUT_RESOLUTION	904	输出分辨率错误
NET_ERR_LAYOUT	905	场景号错误
NET_ERR_INPUT_RESOLUTION	906	输入分辨率不支持
NET_ERR_SUBDEVICE_OFFLINE	907	子设备不在线
NET_ERR_NO_DECODE_CHAN	908	没有空闲解码通道
NET_ERR_MAX_WINDOW_ABILITY	909	开窗能力上限，分布式多屏控制器中解码子设备能力上限或者显示处理器能力上限导致
NET_ERR_ORDER_ERROR	910	调用顺序有误
NET_ERR_PLAYING_PLAN	911	正在执行预案
NET_ERR_DECODER_USED	912	解码板正在使用
NET_ERR_OUTPUT_BOARD_DATA_OVERFLOW	913	输出板数据量超限
NET_ERR_SAME_USER_NAME	914	用户名相同
NET_ERR_INVALID_USER_NAME	915	无效用户名
NET_ERR_MATRIX_USING	916	输入矩阵正在使用
NET_ERR_DIFFERENT_CHAN_TYPE	917	通道类型不同（矩阵输出通道和控制器的输入为不同的类型）
NET_ERR_INPUT_CHAN_BINDED	918	输入通道已经被其他矩阵绑定
NET_ERR_BINDED_OUTPUT_CHAN_OVERFLOW	919	正在使用的矩阵输出通道个数超过矩阵与控制器绑定的通道个数
NET_ERR_MAX_SIGNAL_NUM	920	输入信号源个数达到上限
NET_ERR_INPUT_CHAN_USING	921	输入通道正在使用
NET_ERR_MANAGER_LOGON	922	管理员已经登陆，操作失败
NET_ERR_USERALREADY_LOGON	923	该用户已经登陆，操作失败
NET_ERR_LAYOUT_INIT	924	场景正在初始化，操作失败
NET_ERR_BASEMAP_SIZE_NOT_MATCH	925	底图大小不符
NET_ERR_WINDOW_OPERATING	926	窗口正在执行其他操作，本次操作失败
NET_ERR_SIGNAL_UPLIMIT	927	信号源开窗个数达到上限

6.2 RTSP 通讯库错误码

错误名称	错误值	说明
NET_DVR_RTSP_GETPORTFAILED	407	获取 RTSP 端口错误
NET_DVR_RTSP_DESCRIBESENDTIMEOUT	411	RTSP DESCRIBE 发送超时
NET_DVR_RTSP_DESCRIBESENDERROR	412	RTSP DESCRIBE 发送失败
NET_DVR_RTSP_DESCRIBERECDTIMEOUT	413	RTSP DESCRIBE 接收超时
NET_DVR_RTSP_DESCRIBERECDATALOST	414	RTSP DESCRIBE 接收数据错误
NET_DVR_RTSP_DESCRIBERECDERROR	415	RTSP DESCRIBE 接收失败
NET_DVR_RTSP_DESCRIBESERVERERR	416	RTSP DESCRIBE 服务器返回 401,501 等错误
NET_DVR_RTSP_SETUPSENDTIMEOUT	421	RTSP SETUP 发送超时
NET_DVR_RTSP_SETUPSENDERROR	422	RTSP SETUP 发送错误
NET_DVR_RTSP_SETUPRECDTIMEOUT	423	RTSP SETUP 接收超时
NET_DVR_RTSP_SETUPRECDATALOST	424	RTSP SETUP 接收数据错误
NET_DVR_RTSP_SETUPRECDERROR	425	RTSP SETUP 接收失败
NET_DVR_RTSP_OVER_MAX_CHAN	426	设备超过最大连接数
NET_DVR_RTSP_PLAYSENDTIMEOUT	431	RTSP PLAY 发送超时
NET_DVR_RTSP_PLAYSENDERROR	432	RTSP PLAY 发送错误
NET_DVR_RTSP_PLAYRECDTIMEOUT	433	RTSP PLAT 接收超时
NET_DVR_RTSP_PLAYRECDATALOST	434	RTSP PLAY 接收数据错误
NET_DVR_RTSP_PLAYRECDERROR	435	RTSP PLAY 接收失败
NET_DVR_RTSP_PLAYSERVERERR	436	RTSP PLAY 设备返回错误状态
NET_DVR_RTSP_TEARDOWNSENDTIMEOUT	441	RTSP TEARDOWN 发送超时
NET_DVR_RTSP_TEARDOWNSENDERROR	442	RTSP TEARDOWN 发送错误
NET_DVR_RTSP_TEARDOWNRECDTIMEOUT	443	RTSP TEARDOWN 接收超时
NET_DVR_RTSP_TEARDOWNRECDATALOST	444	RTSP TEARDOWN 接收数据错误
NET_DVR_RTSP_TEARDOWNRECDERROR	445	RTSP TEARDOWN 接收失败
NET_DVR_RTSP_TEARDOWNSERVERERR	446	RTSP TEARDOWN 设备返回错误状态

6.3 软解码库错误码

错误名称	错误值	说明
NET_PLAYM4_NOERROR	500	没有错误
NET_PLAYM4_PARA_OVER	501	输入参数非法
NET_PLAYM4_ORDER_ERROR	502	调用顺序不对
NET_PLAYM4_TIMER_ERROR	503	多媒体时钟设置失败

NET_PLAYM4_DEC_VIDEO_ERROR	504	视频解码失败
NET_PLAYM4_DEC_AUDIO_ERROR	505	音频解码失败
NET_PLAYM4_ALLOC_MEMORY_ERROR	506	分配内存失败
NET_PLAYM4_OPEN_FILE_ERROR	507	文件操作失败
NET_PLAYM4_CREATE_OBJ_ERROR	508	创建线程事件等失败
NET_PLAYM4_CREATE_DDRAW_ERROR	509	创建 directDraw 失败
NET_PLAYM4_CREATE_OFFSCREEN_ERROR	510	创建后端缓存失败
NET_PLAYM4_BUF_OVER	511	缓冲区满，输入流失败
NET_PLAYM4_CREATE_SOUND_ERROR	512	创建音频设备失败
NET_PLAYM4_SET_VOLUME_ERROR	513	设置音量失败
NET_PLAYM4_SUPPORT_FILE_ONLY	514	只能在播放文件时才能使用此接口
NET_PLAYM4_SUPPORT_STREAM_ONLY	515	只能在播放流时才能使用此接口
NET_PLAYM4_SYS_NOT_SUPPORT	516	系统不支持，解码器只能工作在 Pentium 3 以上
NET_PLAYM4_FILEHEADER_UNKNOWN	517	没有文件头
NET_PLAYM4_VERSION_INCORRECT	518	解码器和编码器版本不对应
NET_PLAYM4_INIT_DECODER_ERROR	519	初始化解码器失败
NET_PLAYM4_CHECK_FILE_ERROR	520	文件太短或码流无法识别
NET_PLAYM4_INIT_TIMER_ERROR	521	初始化多媒体时钟失败
NET_PLAYM4_BLT_ERROR	522	位拷贝失败
NET_PLAYM4_UPDATE_ERROR	523	显示 overlay 失败
NET_PLAYM4_OPEN_FILE_ERROR_MULTI	524	打开混合流文件失败
NET_PLAYM4_OPEN_FILE_ERROR_VIDEO	525	打开视频流文件失败
NET_PLAYM4_JPEG_COMPRESS_ERROR	526	JPEG 压缩错误
NET_PLAYM4_EXTRACT_NOT_SUPPORT	527	不支持该文件版本.
NET_PLAYM4_EXTRACT_DATA_ERROR	528	提取文件数据失败

7 结构体说明

7.1 宏定义

宏定义	宏定义值	含义
NAME_LEN	32	用户名长度
PASSWD_LEN	16	密码长度
SERIALNO_LEN	48	序列号长度
MACADDR_LEN	6	MAC 地址长度
MAX_NAMELEN	16	DVR 本地登陆名长度
MAX_DOMAIN_NAME	64	最大域名长度
MAX_CYCLE_CHAN_V30	64	最大轮巡通道数（扩展）
MAX_DAYS	7	每周的天数
MAX_UNITEDMATRIX_NUM	8	级联中最多视频综合平台数量
MAX_OPTICALFIBER_NUM	16	最大光纤条数
MAX_ETHERNET	2	设备可配以太网个数
MAX_DECODECHANNUM	32	最大解码通道数
LOG_INFO_LEN	11840	日志附加信息长度
MAX_ANALOG_ALARMOUT	32	最大模拟报警输出路数
MAX_IP_ALARMOUT	64	允许加入的最多 IP 报警输出路数
MAX_ALARMOUT_V30	96	(MAX_ANALOG_ALARMOUT + MAX_IP_ALARMOUT)
MAX_ALARMOUT_V40	4128	最大报警输出出口（模拟通道+IP 通道）个数扩展
MAX_TIMESEGMENT_V30	8	最大时间段数
IPC_PROTOCOL_NUM	50	IPC 协议最大个数
MAX_EXCEPTIONNUM_V30	32	设备最大异常处理数
MAX_USERNUM_V30	32	设备最大用户数
MAX_ANALOG_CHANNUM	32	最大模拟通道个数
MAX_IP_CHANNEL	32	允许加入的最多 IP 通道数
MAX_CHANNUM_V30	64	(MAX_ANALOG_CHANNUM + MAX_IP_CHANNEL)
MAX_CHANNUM_V40	512	最大通道（模拟通道+IP 通道）个数扩展
MAX_RIGHT	32	设备支持的权限（1-12 表示本地权限，13-32 表示远程权限）
MAX_DISKNUM_V30	33	设备支持的最大硬盘数，最多 33 个硬盘(包括 16 个内置 SATA 硬盘、1 个 eSATA 硬盘和 16 个 NFS 盘)
MAX_PRESET_V30	256	设备支持的最大云台预置点数

MAX_TRACK_V30	256	设备支持的最大云台轨迹数
MAX_CRUISE_V30	256	设备支持的最大云台巡航数
STREAM_ID_LEN	32	流 ID 标识长度
MAX_DISPLAY_NUM	512	最大显示输出个数
MAX_NETWORK_CARD	4	设备可配最大网卡数目
MAX_OSD_LEN	64	输出口 OSD 长度
DESC_LEN	16	云台描述字符串长度
MAX_SERIAL_PORT	8	设备支持的最大 232 串口数
MAX_SCREEN_ADDRESS_LEN	16	特征码最大长度
NAME_LEN	32	用户名长度
PASSWD_LEN	16	密码长度
MATRIX_MAX_OUTPUT_NUM	256	矩阵最大输出通道个数
MAX_WINDOWS_V41	16	最大窗口数(v4.1 版本 SDK)
MAX_DISPNUM_V41	32	最大显示通道个数
MAX_ANALOG_ALARMIN	32	最大 32 路模拟报警输入
MAX_VERSION_LENGTH 8	8	最大版本号长度
MAX_CAM_COUNT	24	摄像机个数

7.2 NET_DVR_ACS_PARAM_TYPE: 门禁主机参数结构体

```
struct{
    DWORD    dwSize;
    DWORD    dwParamType;
    BYTE     byRes[32];
}NET_DVR_ACS_PARAM_TYPE,*LPNET_DVR_ACS_PARAM_TYPE;
```

Members

dwSize

结构体大小

dwParamType

参数类型，按位表示，每位代表一种参数，值：0- 不处理，1- 处理

宏定义	宏定义值	含义
ACS_PARAM_DOOR_STATUS_WEEK_PLAN	0x00000001	门状态周计划参数
ACS_PARAM_VERIFY_WEEK_PALN	0x00000002	读卡器周计划参数
ACS_PARAM_CARD_RIGHT_WEEK_PLAN	0x00000004	卡权限周计划参数
ACS_PARAM_DOOR_STATUS_HOLIDAY_PLAN	0x00000008	门状态假日计划参数
ACS_PARAM_VERIFY_HOLIDAY_PALN	0x00000010	读卡器假日计划参数

ACS_PARAM_CARD_RIGHT_HOLIDAY_PLAN	0x00000020	卡权限假日计划参数
ACS_PARAM_DOOR_STATUS_HOLIDAY_GROUP	0x00000040	门状态假日组参数
ACS_PARAM_VERIFY_HOLIDAY_GROUP	0x00000080	读卡器验证方式假日组参数
ACS_PARAM_CARD_RIGHT_HOLIDAY_GROUP	0x00000100	卡权限假日组参数
ACS_PARAM_DOOR_STATUS_PLAN_TEMPLATE	0x00000200	门状态计划模板参数
ACS_PARAM_VERIFY_PALN_TEMPLATE	0x00000400	读卡器验证方式计划模板参数
ACS_PARAM_CARD_RIGHT_PALN_TEMPLATE	0x00000800	卡权限计划模板参数
ACS_PARAM_CARD	0x00001000	卡参数
ACS_PARAM_GROUP	0x00002000	群组参数
ACS_PARAM_ANTI_SNEAK_CFG	0x00004000	反潜回参数
ACS_PAPAM_EVENT_CARD_LINKAGE	0x00008000	事件及卡号联动参数
ACS_PAPAM_CARD_PASSWD_CFG	0x00010000	密码开门使能参数

byRes

保留，置为 0

Remarks

清空门禁主机参数控制能力，对应 IP 可视对讲主机能力集（接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：[IP_VIEW_DEV_ABILITY](#)）中节点<clearAcsParam>。

See Also

[NET_DVR_RemoteControl](#)

7.3 NET_DVR_AGAIN_RELATEDDEV: 二次确认机关联网络设备参数结构体

```

struct{
    NET\_DVR\_IPADDR    struSIPServer;
    NET\_DVR\_IPADDR    struCenterAddr;
    WORD              wCenterPort;
    BYTE               byRes1[2];
    NET\_DVR\_IPADDR    struIndoorUnit;
    BYTE               byRes[588];
}NET_DVR_AGAIN_RELATEDDEV,*LPNET_DVR_AGAIN_RELATEDDEV;

```

Members

struSIPServer

SIP 服务器 IP 地址

struCenterAddr

中心平台 IP 地址

wCenterPort

中心平台端口

byRes

保留，置为 0

struIndoorUnit

室内主机 IP 地址

byRes

保留，置为 0

See Also

[NET_DVR_VIDEO_INTERCOM_UNIT_RELATEDDEV_UNION](#)

7.4 NET_DVR_ALARMER:报警设备信息

```
struct{
    BYTE    byUserIDValid;
    BYTE    bySerialValid;
    BYTE    byVersionValid;
    BYTE    byDeviceNameValid;
    BYTE    byMacAddrValid;
    BYTE    byLinkPortValid;
    BYTE    byDeviceIPValid;
    BYTE    bySocketIPValid;
    LONG    lUserID;
    BYTE    sSerialNumber[SERIALNO_LEN];
    DWORD   dwDeviceVersion;
    char    sDeviceName[NAME_LEN];
    BYTE    byMacAddr[MACADDR_LEN];
    WORD    wLinkPort;
    char    sDeviceIP[128];
    char    sSocketIP[128];
    BYTE    byIpProtocol;
    BYTE    byRes2[11];
}NET_DVR_ALARMER,*LPNET_DVR_ALARMER;
```

Members

byUserIDValid

userid 是否有效: 0- 无效, 1- 有效

bySerialValid

序列号是否有效: 0- 无效, 1- 有效

byVersionValid

版本号是否有效: 0- 无效, 1- 有效

byDeviceNameValid

设备名字是否有效: 0- 无效, 1- 有效

byMacAddrValid

MAC 地址是否有效: 0- 无效, 1- 有效

byLinkPortValid

Login 端口是否有效: 0- 无效, 1- 有效

byDeviceIPValid

设备 IP 是否有效：0- 无效，1- 有效

bySocketIPValid

Socket IP 是否有效：0-无效，1-有效

lUserID

NET_DVR_Login 或 NET_DVR_Login_V30 返回值，布防时有效

sSerialNumber

序列号

dwDeviceVersion

版本信息：V3.0 以上版本支持的设备最高 8 位为主版本号，次高 8 位为次版本号，低 16 位为修复版本号；V3.0 以下版本支持的设备高 16 位表示主版本，低 16 位表示次版本

sDeviceName

设备名称

byMacAddr

MAC 地址

wLinkPort

设备通讯端口

sDeviceIP

设备 IP 地址

sSocketIP

报警主动上传时的 Socket IP 地址

byIpProtocol

IP 协议：0- IPV4，1- IPV6

byRes2

保留，置为 0

See Also

[NET_DVR_SetDVRMessageCallBack_V30](#)、[NET_DVR_StartListen_V30](#)

7.5 NET_DVR_ALARM_RS485CFG:报警主机 RS485 参数结构体

```
struct{
    DWORD    dwSize;
    BYTE     sDeviceName[NAME_LEN];
    WORD     wDeviceType;
    WORD     wDeviceProtocol;
    DWORD    dwBaudRate;
    BYTE     byDataBit;
    BYTE     byStopBit;
    BYTE     byParity;
    BYTE     byFlowcontrol;
    BYTE     byDuplex;
    BYTE     byWorkMode;
    BYTE     byChannel;
    BYTE     byRes[37];
}
```

```
}NET_DVR_ALARM_RS485CFG, *LPNET_DVR_ALARM_RS485CFG;
```

Members

dwSize

结构体大小

sDeviceName

前端设备名称

wDeviceType

前端设备类型

wDeviceProtocol

前端设备协议，通过获取协议列表 NET_DVR_GetDeviceProtoList 获取

dwBaudRate

波特率(bps), 0-50, 1-75, 2-110, 3-150, 4-300, 5-600, 6-1200, 7-2400, 8-4800, 9-9600, 10-19200, 11-38400, 12-57600, 13-76800, 14-115.2k

byDataBit

数据有几位: 0-5 位, 1-6 位, 2-7 位, 3-8 位

byStopBit

停止位: 0-1 位, 1-2 位

byParity

是否校验: 0-无校验, 1-奇校验, 2-偶校验

byFlowcontrol

是否流控: 0-无, 1-软流控, 2-硬流控

byDuplex

0- 半双工, 1- 全双工

byWorkMode

工作模式: 0- 控制台, 1- 透明通道, 2- 梯控, 3- 读卡器, 0xfe- 自定义(保留), 0xff- 禁用

byChannel

RS485 通道号

byRes

保留

Remarks

对于动环监控主机 V2.0, RS485 参数配置中的 sDeviceName、wDeviceType、wDeviceProtocol 三个字段不能设置（不生效），该配置仅配置 485 串口通信相关参数：波特率（1200~57600）、数据位（8 位或者 7 位加校验位）、停止位（1、2）、检验位（无、奇校验、偶校验）、流控（无）、通信模式（全双工、半双工）、工作模式（控制台、透明通道）。

RS485 参数能力，对应 RS232 和 RS485 串口能力集对应 IP 可视对讲主机能力集（接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：[IP_VIEW_DEV_ABILITY](#)）中节点<RS485>。

See Also

[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)

7.6 NET_DVR_ALARMIN_PARAM:防区参数

```
struct{
    DWORD                dwSize;
    BYTE                 byName[NAME_LEN];
```

```

WORD                wDetectorType;
BYTE                byType;
BYTE                byUploadAlarmRecoveryReport;
DWORD              dwParam;
NET\_DVR\_SCHEDTIME struAlarmTime[MAX_DAYS][MAX_TIMESEGMENT];
BYTE                byAssociateAlarmOut[MAX_ALARMHOST_ALARMOUT_NUM];
BYTE                byAssociateSirenOut[8];
BYTE                bySensitivityParam;
BYTE                byArrayBypass;
BYTE                byJointSubSystem;
BYTE                byModuleStatus;
WORD                wModuleAddress;
BYTE                byModuleChan;
BYTE                byModuleType;
WORD                wZoneIndex;
WORD                wInDelay;
WORD                wOutDelay;
BYTE                byAlarmType;
BYTE                byZoneResistor;
float               fZoneResistorManual;
BYTE                byRes2[32];
}NET_DVR_ALARMIN_PARAM, *LPNET_DVR_ALARMIN_PARAM;

```

Members

dwSize

结构体大小

byName

防区名称

wDetectorType

防区探测器类型，具体定义如下：

```

enum _DETECTOR_TYPE_{
    PANIC_BUTTON = 0,
    MAGNETIC_CONTACT,
    SMOKE_DETECTOR,
    ACTIVE_INFRARED_DETECTOR,
    PASSIVE_INFRARED_DETECTOR,
    GLASS_BREAK_DETECTOR,
    VIBRATION_DETECTOR,
    DUAL_TECHNOLOGY_PIR_DETECTOR,
    TRIPLE_TECHNOLOGY_PIR_DETECTOR,
    HUMIDITY_DETECTOR,
    TEMPERATURE_DETECTOR,
    COMBUSTIBLE_GAS_DETECTOR,
    DYNAMIC_SWITCH,
    CONTROL_SWITCH,

```

```
OTHER_DETECTOR = 0xffff
}DETECTOR_TYPE
```

PANIC_BUTTON

紧急开关

MAGNETIC_CONTACT

门磁开关

SMOKE_DETECTOR

烟感探测器

ACTIVE_INFRARED_DETECTOR

主动红外探测器

PASSIVE_INFRARED_DETECTOR

被动红外探测器

GLASS_BREAK_DETECTOR

玻璃破碎探测器

VIBRATION_DETECTOR

震动探测器

DUAL_TECHNOLOGY_PIR_DETECTOR

双鉴移动探测器

TRIPLE_TECHNOLOGY_PIR_DETECTOR

三技术探测器

HUMIDITY_DETECTOR

湿度探测器

TEMPERATURE_DETECTOR

温感探测器

COMBUSTIBLE_GAS_DETECTOR

可燃气体探测器

DYNAMIC_SWITCH

随动开关

CONTROL_SWITCH

控制开关

OTHER_DETECTOR

其他探测器

byType

防区报警类型，0- 即时防区，1- 24 小时有声防区，2- 延时防区，3- 内部防区，4- 钥匙布撤防防区，5- 火警防区，6- 周界防区，7- 24 小时无声防区，8- 24 小时辅助防区，9- 24 小时震动防区，0xff- 无

byUploadAlarmRecoveryReport

是否上传防区报警恢复报告：0-不上传，1-上传

dwParam

防区参数，延时防区延时多长时间

struAlarmTime

布防时间时间段

byAssociateAlarmOut

防区关联报警输出

byAssociateSirenOut

关联警号输出，数组 0 表示警号 1，数组 1 表示警号 2，以此类推。值为 1 表示输出，0 表示不输出

bySensitivityParam

防区灵敏度参数：0- 10ms，1- 250ms，2- 500ms，3- 750ms

byArrayBypass

是否加入旁路组：0- 不支持组旁路，1- 支持组旁路

byJointSubSystem

防区所属的子系统号，该参数只能获取

byModuleStatus

外接防区模块状态：1- 在线，2- 离线，该参数只能获取

wModuleAddress

模块地址，扩展模块从 1~253，0xFFFF 表示无效，该参数只能获取

byModuleChan

模块通道号，从 1 开始，最大值根据模块类型来决定，0xFF 表示无效，该参数只能获取

byModuleType

模块类型：1- 本地防区，2- 单防区，3- 双防区，4- 8 防区，5- 8 路模拟量防区，6- 单防区触发器

wZoneIndex

防区号，该参数只能获取

wInDelay

进入延时，取值范围：0~255 秒

wOutDelay

退出延时，取值范围：0~255 秒

byAlarmType

报警器类型：0- 无效，1- 常开，2- 常闭

byZoneResistor

防区电阻，单位：千欧，取值：0- 无效，1- 2.2，2- 4.7，3- 5.6，4- 8.2，0xff- 自定义

fZoneResistorManual

防区手动电阻，取值范围：1.0~10.0，精确到小数点后一位，单位：千欧，byZoneResistor 为 0xff 时有效

byRes2

保留

See Also

[NET_DVR_GetDVRConfig](#) 、 [NET_DVR_SetDVRConfig](#)

7.7 NET_DVR_CALL_ROOM_CFG:按键呼叫住户配置结构体

```
struct{
    DWORD    dwSize;
    SHORT    nFloorNumber;
    WORD     wRoomNumber;
    BYTE     byRes[128];
}NET_DVR_CALL_ROOM_CFG,*LPNET_DVR_CALL_ROOM_CFG;
```

Members

dwSize

结构体大小

nFloorNumber

层号

wRoomNumber

房间号

byRes

保留，置为 0

Remarks

按键呼叫住户配置参数能力，对应 IP 可视对讲主机能力集（接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：[IP_VIEW_DEV_ABILITY](#)）中节点<CallRoomConfig>。

See Also

[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDVRConfig](#)

7.8 NET_DVR_CARD_CFG: 卡参数配置结构体

```
struct{
    DWORD                dwSize;
    DWORD                dwModifyParamType;
    BYTE                 byCardNo[ACS_CARD_NO_LEN];
    BYTE                 byCardValid;
    BYTE                 byCardType;
    BYTE                 byLeaderCard;
    BYTE                 byRes1;
    DWORD                dwDoorRight;
    NET\_DVR\_VALID\_PERIOD\_CFG struValid;
    DWORD                dwBelongGroup;
    BYTE                 byCardPassword[CARD_PASSWORD_LEN];
    BYTE                 byCardRightPlan[MAX_DOOR_NUM][MAX_CARD_RIGHT_PLAN_NUM];
    DWORD                dwMaxSwipeTime;
    DWORD                dwSwipeTime;
    WORD                 wRoomNumber;
    WORD                 wFloorNumber;
    BYTE                 byRes2[20];
}NET_DVR_CARD_CFG,*LPNET_DVR_CARD_CFG;
```

Members

dwSize

结构体大小

dwModifyParamType

需要修改的卡参数（设置卡参数时有效），按位表示，每位代表一种参数，值：0- 不修改，1- 需要修改

宏定义	宏定义值	含义
CARD_PARAM_CARD_VALID	0x00000001	卡是否有效参数
CARD_PARAM_VALID	0x00000002	有效期参数

CARD_PARAM_CARD_TYPE	0x00000004	卡类型参数
CARD_PARAM_DOOR_RIGHT	0x00000008	门权限参数
CARD_PARAM_LEADER_CARD	0x00000010	首卡参数
CARD_PARAM_SWIPE_NUM	0x00000020	最大刷卡次数参数
CARD_PARAM_GROUP	0x00000040	所属群组参数
CARD_PARAM_PASSWORD	0x00000080	卡密码参数
CARD_PARAM_RIGHT_PLAN	0x00000100	卡权限计划参数
CARD_PARAM_SWIPED_NUM	0x00000200	已刷卡次数

byCardNo

卡号，特殊卡号定义如下：

0xFFFFFFFFFFFFFFFF: 非法卡号

0xFFFFFFFFFFFFFFFE: 胁迫码

0xFFFFFFFFFFFFFFFD: 超级码

0xFFFFFFFFFFFFFFFC~0xFFFFFFFFFFFFF1: 预留的特殊卡

0xFFFFFFFFFFFFFFF0: 最大合法卡号

byCardValid

卡是否有效：0- 无效，1- 有效（用于删除卡，设置时置为 0 进行删除，获取时此字段始终为 1）

byCardType

卡类型：1- 普通卡（默认），2- 残疾人卡，3- 黑名单卡，4- 巡更卡，5- 胁迫卡，6- 超级卡，7- 来宾卡，8- 解除卡

byLeaderCard

是否为首卡：1- 是，0- 否

byRes1

保留，置为 0

dwDoorRight

门权限，按位表示，从低位到高位表示对门 1~N 是否有权限，值：0- 无权限，1- 有权限

struValid

有效期参数

dwBelongGroup

所属群组，按位表示，从低位到高位表示是否从属群组 1~N，值：0- 不属于，1- 属于

byCardPassword

卡密码

byCardRightPlan

卡权限计划，取值为计划模板编号，同个门不同计划模板采用权限或的方式处理

dwMaxSwipeTime

最大刷卡次数，0 为无次数限制

dwSwipeTime

已刷卡次数

wRoomNumber

房间号

wFloorNumber

层号

byRes2

保留，置为 0

Remarks

卡参数能力，对应门禁主机能力集（接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：ACS_ABILITY）中节点<Card>。

See Also

[NET_DVR_StartRemoteConfig](#)、[NET_DVR_SendRemoteConfig](#)

7.9 NET_DVR_CARD_CFG_COND: 卡参数配置条件结构体

```
struct{
    DWORD    dwSize;
    DWORD    dwCardNum;
    BYTE     byCheckCardNo;
    BYTE     byRes[31];
}NET_DVR_CARD_CFG_COND,*LPNET_DVR_CARD_CFG_COND;
```

Members

dwSize

结构体大小

dwCardNum

设置或获取卡数量，获取时置为 0xffffffff 表示获取所有卡信息

byCheckCardNo

设备是否进行卡号校验：0- 不校验，1- 校验

byRes

保留，置为 0

Remarks

设置卡参数（下发卡参数）时，如果将 byCheckCardNo 置为 0，那么设备将不校验应用层下发的卡号信息，直接写入本地存储，可以一定程度提高卡号下发的速度，但是需要上层应用自己保证卡号信息不重复（整型值不能重复，比如，不能同时含有 1 和 01 这两种卡号）。

See Also

[NET_DVR_StartRemoteConfig](#)

7.10 NET_DVR_CARD_CFG_SEND_DATA: 获取卡参数的发送数据

```
struct{
    DWORD    dwSize;
    BYTE     byCardNo[ACS_CARD_NO_LEN];
    BYTE     byRes[16];
}NET_DVR_CARD_CFG_SEND_DATA,*LPNET_DVR_CARD_CFG_SEND_DATA;
```

Members

dwSize

结构体大小

byCardNo

卡号

byRes

保留，置为 0

See Also

[NET_DVR_SendRemoteConfig](#)

7.11 NET_DVR_CLIENTINFO:预览参数结构体

```
struct{
    LONG      IChannel;
    LONG      ILinkMode;
    HWND      hPlayWnd;
    char      *sMultiCastIP;
}NET_DVR_CLIENTINFO, *LPNET_DVR_CLIENTINFO;
```

Members

IChannel

通道号，1~32 表示模拟通道 1~32，9000 系列混合型 DVR 和 NVR 等设备的 IP 通道从 33 开始

ILinkMode

最高位(31)为 0 表示主码流，为 1 表示子码流；0~30 位表示连接方式：0—TCP 方式，1—UDP 方式，2—多播方式；例如子码流 TCP 连接，则 ILinkMode=0x80000000

hPlayWnd

播放窗口的句柄，为 NULL 表示不显示图像

sMultiCastIP

多播组地址

7.12 NET_DVR_COMPRESSIONCFG_V30:通道压缩参数结构体

```
struct{
    DWORD                                     dwSize;
    NET\_DVR\_COMPRESSION\_INFO\_V30             struNormHighRecordPara;
    NET\_DVR\_COMPRESSION\_INFO\_V30             struRes;
    NET\_DVR\_COMPRESSION\_INFO\_V30             struEventRecordPara;
    NET\_DVR\_COMPRESSION\_INFO\_V30             struNetPara;
}NET_DVR_COMPRESSIONCFG_V30, *LPNET_DVR_COMPRESSIONCFG_V30;
```

Members

dwSize

结构体大小

struRecordPara

录像/事件触发录像的码流压缩参数

struNetPara

网传的码流压缩参数

See Also

[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)

7.13 NET_DVR_COMPRESSION_INFO_V30:码流压缩参数

struct{

```

    BYTE    byStreamType;
    BYTE    byResolution;
    BYTE    byBitrateType;
    BYTE    byPicQuality;
    DWORD   dwVideoBitrate;
    DWORD   dwVideoFrameRate;
    WORD    wIntervalFrameI;
    BYTE    byIntervalBPFrame;
    BYTE    byres1;
    BYTE    byVideoEncType;
    BYTE    byAudioEncType;
    BYTE    byVideoEncComplexity;
    BYTE    byres[9];

```

}NET_DVR_COMPRESSION_INFO_V30, *LPNET_DVR_COMPRESSION_INFO_V30;

Members

byStreamType

码流类型：0-视频流，1-复合流

byResolution

分辨率：0-DCIF(528*384/528*320)，1-CIF(352*288/352*240)，2-QCIF(176*144/176*120)，3-4CIF(704*576/704*480)或 D1(720*576/720*486)，4-2CIF(704*288/704*240)，6-QVGA(320*240)，7-QQVGA(160*120)，12-384*288，13-576*576，16-VGA(640*480)，17-UXGA(1600*1200)，18-SVGA(800*600)，19-HD720P(1280*720)，20-XVGA(1280*960)，21-HD900P(1600*900)，23-1536*1536，24-1920*1920，27-1920*1080p，28-2560*1920，29-1600*304，30-2048*1536，31-2448*2048，32-2448*1200，33-2448*800，34-XGA(1024*768)，35-SXGA(1280*1024)，36-WD1(960*576/960*480)，37-1080i(1920*1080)，38-WXGA(1440*900)，39-HD_F(1920*1080/1280*720)，40-HD_H(1920*540/1280*360)，41-HD_Q(960*540/630*360)，42-2336*1744，43-1920*1456，44-2592*2048，45-3296*2472，46-1376*768，47-1366*768，48-1360*768，49-WSXGA+，50-720*720，51-1280*1280，52-2048*768，53-2048*2048，54-2560*2048，55-3072*2048，56-2304*1296，57-WXGA(1280*800)，58-1600*600，59-1600*900，60-2752*2208，0xff-Auto(使用当前码流分辨率)

byBitrateType

码率类型：0-变码率，1-定码率

byPicQuality

图象质量：0-最好，1-次好，2-较好，3-一般，4-较差，5-差

dwVideoBitrate

码率：0-保留，1-16K(保留)，2-32K，3-48K，4-64K，5-80K，6-96K，7-128K，8-160K，9-192K，10-224K，11-256K，12-320K，13-384K，14-448K，15-512K，16-640K，17-768K，18-896K，19-1024K，20-1280K，

21-1536K, 22-1792K, 23-2048K, 24-3072K, 25-4096K, 26-8192K, 27-16384K。

最高位(31 位)置成 1 表示是自定义码流, 0~30 位表示码流值, 最小值 16k

dwVideoFrameRate

帧率: 0-全部, 1-1/16, 2-1/8, 3-1/4, 4-1/2, 5-1, 6-2, 7-4, 8-6, 9-8, 10-10, 11-12, 12-16, 13-20, 14-15, 15-18, 16-22

wIntervalFrameI

I 帧间隔, 0xffff-无效

byIntervalBPFrame

帧格式: 0-BBP 帧, 1-BP 帧, 2-单 P 帧, 0xff-无效

byres1

保留, 置为 0

byVideoEncType

视频编码类型: 0-私有 264, 1-标准 h264, 2-标准 mpeg4, 7-M-JPEG, 0xff-无效

byAudioEncType

音频编码类型: 0-OggVorbis, 1-G711_U, 2-G711_A, 6-G726, 0xff-无效

byVideoEncComplexity

视频编码复杂度: 0- 低, 1- 中, 2- 高

byres

保留, 置为 0

Remarks

当修改设备视频编码类型时, 需要重启设备生效。

See Also

[NET_DVR_COMPRESSIONCFG_V30](#)

7.14 NET_DVR_CONTROL_GATEWAY: 远程开锁控制参数结构体

```
struct{
    DWORD    dwSize;
    DWORD    dwGatewayIndex;
    BYTE     byCommand;
    BYTE     byRes1[3];
    BYTE     byControlSrc[NAME_LEN];
    BYTE     byRes2[128];
}NET_DVR_CONTROL_GATEWAY, *LPNET_DVR_CONTROL_GATEWAY;
```

Members

dwSize

结构体大小

dwGatewayIndex

门禁序号, 从 1 开始

byCommand

操作命令: 0- 关闭, 1- 打开

byRes1

保留, 置为 0

byControlSrc

操作发起源信息

byRes2

保留，置为 0

Remarks

如果设置了开锁操作发起源信息，那么当开锁事件记录上传时，会同时包含此信息，用于上层应用做识别使用。

远程开锁控制能力，对应 IP 可视对讲主机能力集（接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：[IP_VIEW_DEV_ABILITY](#)）中节点<ControlGateway>。

See Also

[NET_DVR_RemoteControl](#)

7.15 NET_DVR_DAYTIME:时间参数结构体

```
struct{
    BYTE        byHour;
    BYTE        byMinute;
    BYTE        bySecond;
    BYTE        byRes;
    WORD        wMilliSecond;
    BYTE        byRes1[2];
}NET_DVR_DAYTIME, *LPNET_DVR_DAYTIME;
```

Members

byHour

时，取值范围：0~24

byMinute

分，取值范围：0~60

bySecond

秒，取值范围：0~60

byRes

保留，置为 0

wMilliSecond

毫秒，取值范围：0~1000

byRes1

保留，置为 0

See Also

[NET_DVR_SWITCH_DAY_TIME](#)、

7.16 NET_DVR_DDNS_ADDRESS:设备 DDNS 域名信息

```
struct{
    BYTE        byDevAddress[MAX_DOMAIN_NAME];
```

```

BYTE    byDevDdns[MAX_DOMAIN_NAME];
BYTE    byDdnsType;
BYTE    byRes1[3];
WORD    wDevPort;
WORD    wDdnsPort;
BYTE    byres[64];
}NET_DVR_DDNS_ADDRESS, *LPNET_DVR_DDNS_ADDRESS;

```

Members

byDevAddress

设备域名

byDevDdns

DDNS 服务器地址

byDdnsType

域名服务器类型：0- IPServer，1- Dyndns，2- PeanutHull(花生壳)，3- NO-IP，4- hiDDNS

byRes1

保留

wDevPort

设备端口号

wDdnsPort

DDNS 服务器端口

byres

保留

See Also

7.17 NET_DVR_DDNSPARA: 网络应用参数（DDNS）结构体

```

struct{
    BYTE    sUserName[NAME_LEN];
    BYTE    sPassword[PASSWD_LEN];
    BYTE    sDomainName[64];
    BYTE    byEnableDDNS;
    BYTE    res[15];
}NET_DVR_DDNSPARA, *LPNET_DVR_DDNSPARA;

```

Members

sUserName

DDNS 账号用户名

sPassword

DDNS 账号密码

sDomainName

域名

byEnableDDNS

是否使能：0—否，1—是

res

保留，置为 0

See Also

7.18 **NET_DVR_DELAY_TIME**:延时开机参数结构体

```
struct{
    DWORD    dwSize;
    DWORD    dwDelayTime;
    BYTE     byRes[32];
}NET_DVR_DELAY_TIME,*LPNET_DVR_DELAY_TIME;
```

Members

dwSize

结构体大小

dwDelayTime

延时时间，取值范围：0~3000，单位：ms

byRes

保留，置为 0

See Also

[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)

7.19 **NET_DVR_DEVICECFG_V40**:设备参数

```
struct{
    DWORD    dwSize;
    BYTE     sDVRName[NAME_LEN];
    DWORD    dwDVRID;
    DWORD    dwRecycleRecord;
    BYTE     sSerialNumber[SERIALNO_LEN];
    DWORD    dwSoftwareVersion;
    DWORD    dwSoftwareBuildDate;
    DWORD    dwDSPSoftwareVersion;
    DWORD    dwDSPSoftwareBuildDate;
    DWORD    dwPanelVersion;
    DWORD    dwHardwareVersion;
    BYTE     byAlarmInPortNum;
    BYTE     byAlarmOutPortNum;
    BYTE     byRS232Num;
    BYTE     byRS485Num;
    BYTE     byNetworkPortNum;
    BYTE     byDiskCtrlNum;
    BYTE     byDiskNum;
    BYTE     byDVRType;
```

```

BYTE    byChanNum;
BYTE    byStartChan;
BYTE    byDecordChans;
BYTE    byVGANum;
BYTE    byUSBNum;
BYTE    byAuxoutNum;
BYTE    byAudioNum;
BYTE    byIPChanNum;
BYTE    byZeroChanNum;
BYTE    bySupport;
BYTE    byEsataUseage;
BYTE    byIPCPlug;
BYTE    byStorageMode;
BYTE    bySupport1;
WORD    wDevType;
BYTE    byDevTypeName[24];
BYTE    byRes2[16];

```

```
}NET_DVR_DEVICECFG_V40,*LPNET_DVR_DEVICECFG_V40;
```

Members

dwSize

结构体大小

sDVRName

设备名称

dwDVRID

设备 ID 号，用于遥控器，v1.4 的设备号范围为(0-99), v1.5 及以上版本的设备号为(0-255)

dwRecycleRecord

是否循环录像：0—不是；1—是

以下参数不可更改

sSerialNumber

设备序列号

dwSoftwareVersion

软件版本号，V3.0 以上版本支持的设备最高 8 位为主版本号，次高 8 位为次版本号，低 16 位为修复版本号；V3.0 以下版本支持的设备高 16 位表示主版本，低 16 位表示次版本

dwSoftwareBuildDate

软件生成日期，0xYYYYMMDD

dwDSPSoftwareVersion

DSP 软件版本，高 16 位是主版本，低 16 位是次版本

dwDSPSoftwareBuildDate

DSP 软件生成日期，0xYYYYMMDD

dwPanelVersion

前面板版本，高 16 位是主版本，低 16 位是次版本

dwHardwareVersion

硬件版本，高 16 位是主版本，低 16 位是次版本

byAlarmInPortNum

设备模拟报警输入个数

byAlarmOutPortNum

设备模拟报警输出个数

byRS232Num

设备 232 串口个数

byRS485Num

设备 485 串口个数

byNetworkPortNum

网络口个数

byDiskCtrlNum

硬盘控制器个数

byDiskNum

硬盘个数

byDVRType

设备类型，详见“Remarks”说明

byChanNum

设备模拟通道个数

byStartChan

模拟通道的起始通道号

byDecordChans

设备解码路数

byVGANum

VGA 口的个数

byUSBNum

USB 口的个数

byAuxoutNum

辅口的个数

byAudioNum

语音口的个数

byIPChanNum

最大数字通道

byZeroChanNum

零通道编码个数

bySupport

能力，位与结果为 0 表示不支持，1 表示支持

bySupport & 0x1，表示是否支持智能搜索

bySupport & 0x2，表示是否支持备份

bySupport & 0x4，表示是否支持压缩参数能力获取

bySupport & 0x8，表示是否支持双网卡

bySupport & 0x10，表示支持远程 SADP

bySupport & 0x20，表示支持 Raid 卡功能

bySupport & 0x40，表示支持 IPSAN 搜索

bySupport & 0x80，表示支持 rtp over rtsp

byEsataUseage

Esata 的默认用途：0-默认备份，1-默认录像

byIPPlug

0-不支持即插即用，1-支持即插即用

byStorageMode

存储模式：0-盘组模式，1-磁盘配额，2-抽帧模式

bySupport1

能力集扩充，位与结果为 0 表示不支持，1 表示支持

bySupport1 & 0x1, 表示是否支持 snmp v30

bySupport1 & 0x2, 支持区分回放和下载

wDevType

设备型号，详见“Remarks”说明

byDevTypeName

设备型号名称

byRes2

保留，置为 0

Remarks

如果 byDVRType 是 0, 则接口中解析 wDevType 作为设备型号; 如果 byDVRType 非 0, 则接口中 byDVRType 和 wDevType 值相等, 都是 byDVRType。推荐使用 wDevType 作为设备类型。

See Also

[NET_DVR_LOCAL_BYTE_ENCODE_CONVERT](#)、[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)

7.20 NET_DVR_DEVICEINFO_V30:设备参数结构体

```
struct{
    BYTE    sSerialNumber[SERIALNO_LEN];
    BYTE    byAlarmInPortNum;
    BYTE    byAlarmOutPortNum;
    BYTE    byDiskNum;
    BYTE    byDVRType;
    BYTE    byChanNum;
    BYTE    byStartChan;
    BYTE    byAudioChanNum;
    BYTE    byIPChanNum;
    BYTE    byZeroChanNum;
    BYTE    byMainProto;
    BYTE    bySubProto;
    BYTE    bySupport;
    BYTE    bySupport1;
    BYTE    bySupport2;
    WORD    wDevType;
    BYTE    bySupport3;
    BYTE    byMultiStreamProto;
    BYTE    byStartDChan;
    BYTE    byStartDTalkChan;
```

```

BYTE    byHighDChanNum;
BYTE    bySupport4;
BYTE    byLanguageType;
BYTE    byRes2[9];

```

```
}NET_DVR_DEVICEINFO_V30,*LPNET_DVR_DEVICEINFO_V30;
```

Members

sSerialNumber

序列号

byAlarmInPortNum

报警输入个数

byAlarmOutPortNum

报警输出个数

byDiskNum

硬盘个数

byDVRType

设备类型，详见“Remarks”说明

byChanNum

设备模拟通道个数

byStartChan

起始通道号，目前设备通道号从 1 开始

byAudioChanNum

设备语音通道数

byIPChanNum

设备最大数字通道个数

byZeroChanNum

零通道编码个数

byMainProto

主码流传输协议类型：0-private，1-rtsp

bySubProto

子码流传输协议类型：0-private，1-rtsp

bySupport

能力，位与结果为 0 表示不支持，1 表示支持

bySupport & 0x1，表示是否支持智能搜索

bySupport & 0x2，表示是否支持备份

bySupport & 0x4，表示是否支持压缩参数能力获取

bySupport & 0x8，表示是否支持双网卡

bySupport & 0x10，表示支持远程 SADP

bySupport & 0x20，表示支持 Raid 卡功能

bySupport & 0x40，表示支持 IPSAN 目录查找

bySupport & 0x80，表示支持 rtp over rtsp

bySupport1

能力集扩充，位与结果为 0 表示不支持，1 表示支持

bySupport1 & 0x1，表示是否支持 snmp v30

bySupport1 & 0x2，表示是否支持区分回放和下载

bySupport1 & 0x4, 表示是否支持布防优先级

bySupport2

能力集扩充, 位与结果为 0 表示不支持, 1 表示支持

bySupport2 & 0x1, 表示解码器是否支持通过 URL 取流解码

bySupport2 & 0x2, 表示是否支持 FTPV40

bySupport2 & 0x4, 表示是否支持 ANR(断网录像)

wDevType

设备型号, 详见“Remarks”说明

bySupport3

能力集扩展, 位与结果: 0- 不支持, 1- 支持

bySupport3 & 0x1, 表示是否支持多码流

bySupport3 & 0x4, 表示是否支持按组配置, 具体包含通道图像参数、报警输入参数、IP 报警输入/输出接入参数、用户参数、设备工作状态、JPEG 抓图、定时和时间抓图、硬盘盘组管理等

bySupport3 & 0x20, 表示是否支持通过 DDNS 域名解析取流

byMultiStreamProto

是否支持多码流, 按位表示, 位与结果: 0-不支持, 1-支持

byMultiStreamProto & 0x1, 表示是否支持码流 3

byMultiStreamProto & 0x2, 表示是否支持码流 4

byMultiStreamProto & 0x40, 表示是否支持主码流

byMultiStreamProto & 0x80, 表示是否支持子码流

byStartDChan

起始数字通道号, 0 表示无数字通道, 比如 DVR 或 IPC

byStartDTalkChan

起始数字对讲通道号, 区别于模拟对讲通道号, 0 表示无数字对讲通道

byHighDChanNum

数字通道个数, 高 8 位

bySupport4

能力集扩展, 按位表示, 位与结果: 0- 不支持, 1- 支持

bySupport4 & 0x4 表示是否支持拼控统一接口

byLanguageType

支持语种能力, 按位表示, 位与结果: 0- 不支持, 1- 支持

byLanguageType == 0, 表示老设备, 不支持该字段

byLanguageType & 0x1, 表示是否支持中文

byLanguageType & 0x2, 表示是否支持英文

byRes2

保留, 置为 0

Remarks

如果 byDVRType 是 0, 则接口中解析 wDevType 作为设备型号; 如果 byDVRType 非 0, 则接口中 byDVRType 和 wDevType 值相等, 都是 byDVRType。推荐使用 wDevType 作为设备类型。

See Also

[NET DVR Login V30](#)

7.21 NET_DVR_DIRECT_CONNECT_CHAN_INFO:IP 通道信息

```

struct{
    BYTE                byEnable;
    BYTE                byProType;
    BYTE                byZeroChan;
    BYTE                byPriority;
    BYTE                sUserName[NAME_LEN];
    BYTE                sPassword[PASSWD_LEN];
    BYTE                byDomain[MAX_DOMAIN_NAME];
    NET_DVR_IPADDR      struIP;
    WORD                wDVRPort;
    BYTE                byStreamType;
    BYTE                byOnline;
    DWORD               dwChannel;
    BYTE                byTransProtocol;
    BYTE                byLocalBackUp;
    WORD                wDirectLastTime;
    BYTE                byChanNo[CHAN_NO_LEN];
}NET_DVR_DIRECT_CONNECT_CHAN_INFO, *LPNET_DVR_DIRECT_CONNECT_CHAN_INFO;

```

Members

byEnable

是否启用

byProType

协议类型: 0- 私有协议(default), 1- 松下协议, 2- 索尼, 更多协议通过 [NET_DVR_GetIPCProtoList_V41](#) 获取

byZeroChan

是否是零通道: 0- 不是, 1- 是

byPriority

优先级, 取值范围: 1~5

sUserName

用户名

sPassword

密码

byDomain

设备域名

struIP

IP 地址

wDVRPort

端口号

byStreamType

码流类型: 0- 主码流, 1- 子码流

byOnline

在线状态（只读）：0- 不在线，1- 在线

dwChannel

通道号

byTransProtocol

协议类型：0- TCP，1- UDP，2- 多播

byLocalBackUp

本地备份：0- 不启用 CVR 本地备份，1- 启用 CVR 本地备份（即录像时的流一份保存在录像卷，一份保存在存档卷）

wDirectLastTime

导播持续时间，单位：s，取值范围通过能力集获取

byChanNo

通道编号，用于 VAG 取流

See Also

[NET_DVR_STREAM_TYPE_UNION](#)

7.22 NET_DVR_DOOR_CFG 门参数配置结构体

```
struct{
    DWORD    dwSize;
    BYTE     byDoorName[DOOR_NAME_LEN];
    BYTE     byMagneticType;
    BYTE     byOpenButtonType;
    BYTE     byOpenDuration;
    BYTE     byDisabledOpenDuration;
    BYTE     byMagneticAlarmTimeout;
    BYTE     byEnableDoorLock;
    BYTE     byEnableLeaderCard;
    BYTE     byRes1;
    DWORD    dwLeaderCardOpenDuration;
    BYTE     byStressPassword[STRESS_PASSWORD_LEN];
    BYTE     bySuperPassword[SUPER_PASSWORD_LEN];
    BYTE     byUnlockPassword[UNLOCK_PASSWORD_LEN];
    BYTE     byRes2[56];
}NET_DVR_DOOR_CFG,*LPNET_DVR_DOOR_CFG;
```

Members

dwSize

结构体大小

byDoorName

门名称

byMagneticType

门磁类型：0- 常闭，1- 常开

byOpenButtonType

开门按钮类型：0- 常闭，1- 常开

byOpenDuration

开门持续时间，取值范围：1~255s

byDisabledOpenDuration

残疾人卡开门持续时间，取值范围：1~255s

byMagneticAlarmTimeout

门磁检测超时报警时间，取值范围：0~255s，0 表示不报警

byEnableDoorLock

是否启用闭门回锁：0- 否，1- 是

byEnableLeaderCard

是否启用首卡常开功能：0- 否，1- 是

byRes1

保留，置为 0

dwLeaderCardOpenDuration

首卡常开持续时间，取值范围：1~1440，单位：min（分钟）

byStressPassword

胁迫密码

bySuperPassword

超级密码

byUnlockPassword

解除码，解锁密码

byRes2

保留，置为 0

Remarks

门参数能力，对应 IP 可视对讲主机能力集（接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：[IP_VIEW_DEV_ABILITY](#)）中节点<Door>。

See Also

[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)

7.23 NET_DVR_ELEVATORCONTROL_CFG_V40:可视对讲梯控器参数配置结构体

```
struct{
    DWORD          dwSize;
    BYTE           byEnable;
    BYTE           byRes1;
    BYTE           byInterfaceType;
    BYTE           byRS485Protocol;
    BYTE           byNetworkType;
    BYTE           byRes2;
    WORD           wServerPort;
    NET\_DVR\_IPADDR struServerIP;
    BYTE           byRes[256];
}
```

```
}NET_DVR_ELEVATORCONTROL_CFG_V40, *LPNET_DVR_ELEVATORCONTROL_CFG_V40;
```

Members

dwSize

结构体大小

byEnable

是否启用梯控：0- 否，1- 是

byRes1

保留，置为 0

byInterfaceType

接口类型：0- 无，1- RS485，2- 网口

byRS485Protocol

RS485 协议类型：0- 未知，1- 私有，0xff- 自定义，当 byInterfaceType 取值为 1 时有效

byNetworkType

网口协议类型：0- 未知，1- 私有，0xff- 自定义，当 byInterfaceType 取值为 2 时有效

byRes2

保留，置为 0

wServerPort

梯控服务器端口号，当网卡协议类型为“1- 私有”时有效

struServerIP

梯控服务器 IP 地址，当网卡协议类型为“1- 私有”时有效

byRes

保留，置为 0

Remarks

梯控器参数能力，对应 IP 可视对讲主机能力集（接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：[IP_VIEW_DEV_ABILITY](#)）中节点<ElevatorControl>。

See Also

[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDVRConfig](#)

7.24 NET_DVR_ETHERNET_V30:以太网配置参数

```
struct{
```

[NET_DVR_IPADDR](#) struDVRIP;

[NET_DVR_IPADDR](#) struDVRIPMask;

DWORD dwNetInterface;

WORD wDVRPort;

WORD wMTU;

BYTE byMACAddr[MACADDR_LEN];

BYTE byRes[2];

```
}NET_DVR_ETHERNET_V30, *LPNET_DVR_ETHERNET_V30;
```

Members

struDVRIP

设备 IP 地址

struDVRIPMask

设备 IP 地址掩码

dwNetInterface

网络接口: 1-10MBase-T; 2-10MBase-T 全双工; 3-100MBase-TX; 4-100M 全双工; 5-10M/100M/1000M 自适应; 6-1000M 全双工

wDVRPort

设备端口号

wMTU

MTU 设置, 默认 1500

byMACAddr

设备物理地址

byRes

保留

Remarks

MTU 的设置范围为 500-9676, 若 MTU 设置过小客户端将无法注册到设备, 并且客户端预览、回放、配置参数也会失败。

See Also

[NET_DVR_NETCFG_V30](#)、

7.25 NET_DVR_EXCEPTION_V30:异常参数

struct{

DWORD

dwSize;

[NET_DVR_HANDLEEXCEPTION_V30](#)

struExceptionHandleType[MAX_EXCEPTIONNUM_V30];

}NET_DVR_EXCEPTION_V30,*LPNET_DVR_EXCEPTION_V30;

Members

dwSize

结构体大小

struExceptionHandleType

异常信息处理方式:

数组 0—硬盘满

数组 1—硬盘出错

数组 2—网线断

数组 3—IP 地址冲

数组 4—非法访问

数组 5—输入/输出视频制式不匹配

数组 6—视频信号异常

数组 7—录像异常

数组 8—阵列异常

数组 9—前端/录像分辨率不匹配异常

数组 10—行车超速 (车载专用)

数组 11—热备异常 (N+1 使用)

数组 12—温度异常

数组 13—子系统异常

数组 14—风扇异常

See Also

[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)

7.26 NET_DVR_EXCEPTION_V40: 异常参数配置（扩展）

```
struct{
    DWORD dwSize;
    DWORD dwMaxGroupNum;
    NET\_DVR\_HANDLEEXCEPTION\_V41 struExceptionHandle[MAX_EXCEPTIONNUM_V30];
    BYTE byRes[128];
}NET_DVR_EXCEPTION_V40,*LPNET_DVR_EXCEPTION_V40;
```

Members

dwSize

结构体大小

dwMaxGroupNum

设备支持的异常类型最大组数（只读）

struExceptionHandle

异常信息处理方式，数组的每个元素都表示一种异常类型：

数组 0—硬盘满

数组 1—硬盘出错

数组 2—网线断

数组 3—IP 地址冲突

数组 4—非法访问

数组 5—输入/输出视频制式不匹配

数组 6—视频信号异常

数组 7—录像异常

数组 8—阵列异常

数组 9—前端/录像分辨率不匹配异常

数组 10-行车超速（车载专用）

数组 11-热备异常（N+1 使用）

数组 12-温度异常

数组 13-子系统异常

数组 14-风扇异常

数组 15-POE 供电异常

byRes

保留，置为 0

See Also

[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)

7.27 NET_DVR_FTPCFG_V40: FTP 配置结构体

```
struct{
```

```

NET_DVR_STRUCTHEAD          struStruceHead;
BYTE                          byEnableFTP;
BYTE                          byProtocolType;
WORD                          wFTPPort;
union{
    struct{
        BYTE                  szDomain[64];
        BYTE                  byRes1[80];
    }struDomain;
    struct{
        NET_DVR_IPADDR      strulp;
    }struAddrIP;
}unionServer;
BYTE                          szUserName[32];
BYTE                          szPassWORD[16];
BYTE                          szTopCustomDir[64];
BYTE                          szSubCustomDir[64];
BYTE                          byDirLevel;
BYTE                          byTopDirMode;
BYTE                          bySubDirMode;
BYTE                          byType;
BYTE                          byEnableAnony;
BYTE                          byAddresType;
BYTE                          byFTPPicType;
BYTE                          byPicArchivingInterval;
NET_DVR_PICTURE_NAME_EX      struPicNameRule;
BYTE                          byPicNameRuleType;
BYTE                          byRes[203];
}NET_DVR_FTPCFG_V40,*LPNET_DVR_FTPCFG_V40;

```

Members

struStruceHead

结构体的头信息

byEnableFTP

是否启动 ftp 上传功能: 0- 否, 1- 是

byProtocolType

协议类型: 0- FTP, 1- SFTP

wFTPPort

FTP 端口号

szDomain

服务器地址, 可以使 IPv4、IPv6 或是域名

strulp

IPv4 或者 IPv6 地址

szUserName

用户名

szPassWORD

密码

szTopCustomDir

自定义一级目录

szSubCustomDir

自定义二级目录

byDirLevel

0- 不使用目录结构, 直接保存在根目录; 1- 使用 1 级目录; 2- 使用 2 级目录

byTopDirMode

一级目录: 0x1- 使用设备名, 0x2- 使用设备号, 0x3- 使用设备 ip 地址, 0x4- 使用监测点, 0x5- 使用时间(年月), 0x6- 使用自定义, 0x7- 违规类型, 0x8- 方向, 0x9- 地点, 0xa- 栋号单元号, 0xb- 期号

bySubDirMode

二级目录: 0x1- 使用通道名, 0x2- 使用通道号, 0x3- 使用时间(年月日), 0x4- 使用车道号, 0x5- 使用自定义, 0x6- 违规类型, 0x7- 方向, 0x8- 地点, 0x9- 车位编号, 0xa- 栋号单元号, 0xb- 门口机编号, 0xc- 设备编号

byType

FTP 类型: 0-主服务器, 1-备用服务器

byEnableAnony

启用匿名: 0- 否, 1- 是

byAddrType

地址类型: 0- 使用 IPV4 或者 IPV6, 1- 使用域名

byFTPPicType

FTP 上传图片类型: 0- 保留, 1- 抓拍图片命名规则

byPicArchivingInterval

图片归档间隔, 取值范围: [0,30], 0 表示关闭

struPicNameRule

图片命名规则, byFTPPicType 为 1 时有效

byPicNameRuleType

图片命名规则类型: 0- 默认类型, 1- 图片前缀名定义(启用 struPicNameRule 中的 byPicNamePrefix 字段)

byRes

保留, 置为 0

See Also

[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDeviceConfig](#)

7.28 NET_DVR_FTP_TYPE: FTP 类型信息结构体

```
struct{
    BYTE    byType;
    BYTE    byRes[3];
}NET_DVR_FTP_TYPE,*LPNET_DVR_FTP_TYPE;
```

Members

byType

0-首选 FTP 服务器, 1-备用 FTP 服务器

byRes

保留

See Also

[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDeviceConfig](#)

7.29 NET_DVR_HANDLEEXCEPTION:报警和异常处理结构体

```
struct{
    DWORD    dwHandleType;
    BYTE      byRelAlarmOut[MAX_ALARMOUT];
}NET_DVR_HANDLEEXCEPTION, *LPNET_DVR_HANDLEEXCEPTION;
```

Members

dwHandleType

处理方式:

0x00: 无响应

0x01: 监视器上警告

0x02: 声音警告

0x04: 上传中心

0x08: 触发报警输出

0x10: Jpeg 抓图并上传 EMail

byRelAlarmOut

报警触发的输出通道, 0-不触发, 1-触发输出

See Also

[NET_DVR_VIHOST](#)、[NET_DVR_MOTION](#)、[NET_DVR_HIDEALARM](#)

7.30 NET_DVR_HANDLEEXCEPTION_V30:报警和异常处理

```
struct{
    DWORD    dwHandleType;
    BYTE      byRelAlarmOut[MAX_ALARMOUT_V30];
}NET_DVR_HANDLEEXCEPTION_V30, *LPNET_DVR_HANDLEEXCEPTION_V30;
```

Members

dwHandleType

处理方式:

0x00: 无响应

0x01: 监视器上警告

0x02: 声音警告

0x04: 上传中心

0x08: 触发报警输出

0x10: Jpeg 抓图并上传 Email

0x20: 无线声光报警器联动

0x40: 联动电子地图(目前仅 PCNVR 支持)

0x200: 抓图并上传 ftp

byRelAlarmOut

报警触发的输出通道，0-不触发，1-触发输出

See Also

[NET_DVR_EXCEPTION_V30](#)

7.31 NET_DVR_HANDLEEXCEPTION_V41:异常参数信息结构体

```
struct{
    DWORD    dwHandleType;
    DWORD    dwMaxAlarmOutChannelNum;
    DWORD    dwRelAlarmOut[MAX_ALARMOUT_V40];
    BYTE     byRes[64];
}NET_DVR_HANDLEEXCEPTION_V41,*LPNET_DVR_HANDLEEXCEPTION_V41;
```

Members

dwHandleType

处理方式，各种异常处理方式的"或"结果，异常处理方式：

0x00: 无响应

0x01: 监视器上警告

0x02: 声音警告

0x04: 上传中心

0x08: 触发报警输出

0x10: Jpeg 抓图并上传 EMail

0x20: 无线声光报警器联动

0x40: 联动电子地图(目前仅 PCNVR 支持)

0x200: 抓图并上传 ftp E.g. dwHandleType==0x01|0x04 表示配置报警发生时联动监视器上警告并且将报警信息上传中心

0x400: 虚交侦测 联动 聚焦模式（提供可配置项，原先设备自动完成）IPC5.1.0

0x800: PTZ 联动跟踪(球机跟踪目标)

dwMaxAlarmOutChannelNum

设备最大支持的触发报警输出通道数（只读）

dwRelAlarmOut

触发报警输出通道，数组元素值表示报警输出号（从 0 开始），例如：dwRelAlarmOut[i]==3 表示触发第 4 个报警输出通道。中间遇到 0xffffffff 则后续无效。

byRes

保留，置为 0

See Also

[NET_DVR_EXCEPTION_V40](#)

7.32 NET_DVR_HIDEALARM:遮挡报警参数结构体

```
struct{
    DWORD    dwEnableHideAlarm;
    WORD     wHideAlarmAreaTopLeftX;
```

```

WORD                wHideAlarmAreaTopLeftY;
WORD                wHideAlarmAreaWidth;
WORD                wHideAlarmAreaHeight;
NET\_DVR\_HANDLEEXCEPTION strHideAlarmHandleType;
NET\_DVR\_SCHEDTIME      struAlarmTime[MAX_DAYS][MAX_TIMESEGMENT];
}NET_DVR_HIDEALARM, *LPNET_DVR_HIDEALARM;

```

Members

dwEnableHideAlarm
是否启动遮挡报警：0-否，1-低灵敏度，2-中灵敏度，3-高灵敏度

wHideAlarmAreaTopLeftX
遮挡区域的 x 坐标

wHideAlarmAreaTopLeftY
遮挡区域的 y 坐标

wHideAlarmAreaWidth
遮挡区域的宽

wHideAlarmAreaHeight
遮挡区域的高

strHideAlarmHandleType
处理方法

struAlarmTime
布防时间

Remarks

SDK 设定整个图像的区域大小为 704*576，结构中遮挡区域的坐标以及宽和高需要转换成 704*576 大小区域下的坐标和宽高值。

See Also

[NET_DVR_PICCFG](#)

7.33 **NET_DVR_HOLIDAY_PARAM_CFG**:假日参数配置结构体

```

struct{
    DWORD                dwSize;
    NET\_DVR\_HOLIDAY\_PARAM struHolidayParam[MAX_HOLIDAY_NUM];
    DWORD                byRes[40];
}NET_DVR_HOLIDAY_PARAM_CFG, *LPNET_DVR_HOLIDAY_PARAM_CFG;

```

Members

dwSize
结构体大小

struHolidayParam
假日参数

byRes
保留

See Also

[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)

7.34 NET_DVR_HOLIDAY_PARAM:假日参数结构体

```
struct{
    BYTE                byEnable;
    BYTE                byDateMode;
    BYTE                byRes1[2];
    NET\_DVR\_HOLIDATE\_UNION uHolidate;
    BYTE                byName[NAME_LEN];
    BYTE                byRes2[20];
}NET_DVR_HOLIDAY_PARAM, *LPNET_DVR_HOLIDAY_PARAM;
```

Members

byEnable
是否启用

byDateMode
日期模式:0-模式 A,1-模式 B,2-模式 C

byRes1
保留

uHolidate
假日日期

byName
假日名称

byRes2
保留

See Also

[NET_DVR_HOLIDAY_PARAM_CFG](#)

7.35 NET_DVR_HOLIDATE_UNION:假日日期结构联合体

```
union{
    DWORD                dwSize[3];
    NET\_DVR\_HOLIDATE\_MODEA struModeA;
    NET\_DVR\_HOLIDATE\_MODEB struModeB;
    NET\_DVR\_HOLIDATE\_MODEC struModeC;
}NET_DVR_HOLIDATE_UNION, *LPNET_DVR_HOLIDATE_UNION;
```

Members

dwSize
联合体大小

struModeA
日期模式 A

struModeB
日期模式 B

struModeC
日期模式 C

See Also

[NET_DVR_HOLIDAY_PARAM](#)

7.36 NET_DVR_HOLIDATE_MODEA:假日日期模式 A

```
struct{
    BYTE    byStartMonth;
    BYTE    byStartDay;
    BYTE    byEndMonth;
    BYTE    byEndDay;
    BYTE    byRes[4];
}NET_DVR_HOLIDATE_MODEA, *LPNET_DVR_HOLIDATE_MODEA;
```

Members

byStartMonth
开始月，从 1 开始

byStartDay
开始日，从 1 开始

byEndMonth
结束月份

byEndDay
结束日

byRes
保留

See Also

[NET_DVR_HOLIDATE_UNION](#)

7.37 NET_DVR_HOLIDATE_MODEB:假日日期模式 B

```
struct{
    BYTE    byStartMonth;
    BYTE    byStartWeekNum;
    BYTE    byStartWeekday;
    BYTE    byEndMonth;
    BYTE    byEndWeekNum;
    BYTE    byEndWeekday;
    BYTE    byRes[2];
}NET_DVR_HOLIDATE_MODEB, *LPNET_DVR_HOLIDATE_MODEB;
```

Members

byStartMonth
开始月份，从 1 开始

byStartWeekNum
第几个星期,从 1 开始

byStartWeekday

星期几

byEndMonth

结束月份, 从 1 开始

byEndWeekNum

第几个星期, 从 1 开始

byEndWeekday

星期几

byRes

保留

See Also

[NET_DVR_HOLIDATE_UNION](#)

7.38 NET_DVR_HOLIDATE_MODEC: 假日日期模式 c

```
struct{
    WORD        wStartYear;
    BYTE        byStartMon;
    BYTE        byStartDay;
    WORD        wEndYear;
    BYTE        byEndMon;
    BYTE        byEndDay;
}NET_DVR_HOLIDATE_MODEC, *LPNET_DVR_HOLIDATE_MODEC;
```

Members

wStartYear

开始年份

byStartMon

开始月份, 从 1 开始

byStartDay

开始日

wEndYear

结束年份

byEndMon

结束月份

byEndDay

结束日

See Also

[NET_DVR_HOLIDATE_UNION](#)

7.39 NET_DVR_IP_ADDRESS: 设备 IP 地址信息

```
struct{
    BYTE        byDevAddress[MAX_DOMAIN_NAME];
    WORD        wDevPort;
```

```

    BYTE    byres[134];
}NET_DVR_IP_ADDRESS, *LPNET_DVR_IP_ADDRESS;

```

Members

byDevAddress

设备 IP 地址

wDevPort

设备端口号

byres

保留

See Also

[NET_DVR_MATRIX_DEC_REMOTE_PLAY_EX](#)

7.40 NET_DVR_IPADDR:IP 地址

```

struct{
    char    slpV4[16];
    BYTE    slpV6[128];
}NET_DVR_IPADDR, *LPNET_DVR_IPADDR;

```

Members

slpV4

设备 IPv4 地址

slpV6

设备 IPv6 地址

See Also

[NET_DVR_PPPCFG_V30](#)、[NET_DVR_ETHERNET_V30](#)、[NET_DVR_NETCFG_V30](#)、
[NET_DVR_LOG_V30](#)、[NET_DVR_USER_INFO_V40](#)

7.41 NET_DVR_LOCAL_ABILITY_PARSE_CFG:能力集解析库配置

```

struct{
    BYTE    byEnableAbilityParse;
    BYTE    byRes[127];
}NET_DVR_LOCAL_ABILITY_PARSE_CFG,*LPNET_DVR_LOCAL_ABILITY_PARSE_CFG;

```

Members

byEnableAbilityParse

使用能力集解析库：0-不使用，1-使用，默认不使用

byRes

保留，置为 0

Remarks

模拟能力集默认禁用，调用该接口可以启用模拟能力集，支持获取设备各种能力。如果需要获取能力集（NET_DVR_GetDeviceAbility），可以调用此接口来启用模拟能力集，并且需要加载 LocalXml.zip（要求和 SDK 库文件放在同一个目录下）。

See Also

[NET_DVR_GetSDKLocalCfg](#)、[NET_DVR_SetSDKLocalCfg](#)

7.42 NET_DVR_LOCAL_BYTE_ENCODE_CONVERT: 字符编码转换参数结构体

```
struct{
    CHAR_ENCODE_CONVERT    fnCharConvertCallBack;
    BYTE                    byRes[256];
}NET_DVR_LOCAL_BYTE_ENCODE_CONVERT,*LPNET_DVR_LOCAL_BYTE_ENCODE_CONVERT;
```

Members

fnCharConvertCallBack

字符编码转换回调函数

byRes

保留，置为 0

Callback Function

```
typedef int(CALLBACK *CHAR_ENCODE_CONVERT)(
    char        *pInput,
    DWORD       dwInputLen,
    DWORD       dwInEncodeType,
    char        *pOutput,
    DWORD       dwOutputLen,
    DWORD       dwOutEncodeType
);
```

Callback Function Parameters

pInput

[in] 输入字符串，内存由 SDK 申请，字符串数据也由 SDK 提供

dwInputLen

[in] 输入字符串缓冲区大小

dwInEncodeType

[in] 输入的字符编码格式：0- 无字符编码信息(老设备)，1- GB2312(简体中文)，2- GBK，3- BIG5(繁体中文)，4- Shift_JIS(日文)，5- EUC-KR(韩文)，6- UTF-8，7- ISO8859-1，8- ISO8859-2，9- ISO8859-3，...，依次类推，21- ISO8859-15(西欧)

pOutput

[out] 输出字符串，内存由 SDK 申请，存放使用用户字符编码接口转换之后的字符串

dwOutputLen

[out] 输出字符串缓冲区大小

dwOutEncodeType

[out] 输出字符编码格式：0- 无字符编码信息(老设备)，1- GB2312(简体中文)，2- GBK，3- BIG5(繁体中文)，4- Shift_JIS(日文)，5- EUC-KR(韩文)，6- UTF-8，7- ISO8859-1，8- ISO8859-2，9- ISO8859-3，...，依次类推，21- ISO8859-15(西欧)

Remarks

- 回调函数的返回值：-1 表示失败，0 表示成功（内存足够存放转换以后的字符串）。
- 设备的字符编码类型在登录接口返回，对应 [NET_DVR_DEVICEINFO_V40](#) 结构体中的参数

byCharEncodeType。SDK 内部需要字符编码转换时，SDK 默认使用 libiconv 库进行类型转换。如果不想使用 libiconv 编码库，可以调用 NET_DVR_SetSDKLocalCfg(类型: NET_SDK_LOCAL_CFG_TYPE_BYTE_ENCODE) 设置字符转码回调函数，告知 SDK 用户自己的字符编码接口，然后 SDK 将使用用户提供的字符编码接口进行字符串处理。

See Also

[NET_DVR_SetSDKLocalCfg](#)

7.43 NET_DVR_LOCAL_CHECK_DEV:设备在线巡检参数结构体。

```
struct{
    DWORD    dwCheckOnlineTimeout;
    DWORD    dwCheckOnlineNetFailMax;
    BYTE     byRes[256];
}NET_DVR_LOCAL_CHECK_DEV, *LPNET_DVR_LOCAL_CHECK_DEV;
```

Members

dwCheckOnlineTimeout

巡检时间间隔，单位：ms，取值范围：30s~120s，0 表示用默认值(120s)，推荐设置 30s

dwCheckOnlineNetFailMax

由于网络原因失败的最大累加次数，达到该次数，SDK 才回调用户异常消息，0 表示使用默认值 1，推荐设置 3 次

byRes

保留，置为 0

Remarks

- SDK 按照该结构体中的时间间隔对设备进行自动巡检，巡检过程中如果连失败或者重连成功在 [NET_DVR_SetExceptionCallBack V30](#) 设置的异常消息回调函数中返回，对应异常消息类型为：EXCEPTION_EXCHANGE、RESUME_EXCHANGE。
- 推荐设置 30s 时间间隔、3 次，即心跳间隔为 1.5 分钟。

See Also

[NET_DVR_GetSDKLocalCfg](#)、[NET_DVR_SetSDKLocalCfg](#)

7.44 NET_DVR_LOCAL_MEM_POOL_CFG:内存池本地配置

```
struct{
    DWORD    dwAlarmMaxBlockNum;
    DWORD    dwAlarmReleaseInterval;
    BYTE     byRes[60];
}NET_DVR_LOCAL_MEM_POOL_CFG, *LPNET_DVR_LOCAL_MEM_POOL_CFG;
```

Members

dwAlarmMaxBlockNum

报警模块内存池最多向系统申请的内存块（block）个数，每个 block 为 64MB，超过这个上限则不向系统申请，0 表示无上限

dwAlarmReleaseInterval

报警模块空闲内存释放的间隔，单位：秒，为 0 表示不释放空闲的内存

byRes

保留，置为 0

See Also

[NET_DVR_GetSDKLocalCfg](#)、[NET_DVR_SetSDKLocalCfg](#)

7.45 NET_DVR_LOCAL_MODULE_RECV_TIMEOUT_CFG:按模块配置超时时间

```
struct{
    DWORD    dwPreviewTime;
    DWORD    dwAlarmTime;
    DWORD    dwVodTime;
    DWORD    dwElse;
    BYTE     byRes[512];
}NET_DVR_LOCAL_MODULE_RECV_TIMEOUT_CFG,*LPNET_DVR_LOCAL_MODULE_RECV_TIMEOUT_CFG;
```

Members

dwPreviewTime

预览模块超时时间，单位：毫秒，取值范围：0~3000,000，0 表示恢复默认设置

dwAlarmTime

报警模块超时时间，单位：毫秒，取值范围：0~3000,000，0 表示恢复默认设置

dwVodTime

回放模块超时时间，单位：毫秒，取值范围：0~3000,000，0 表示恢复默认设置

dwElse

其他模块超时时间，单位：毫秒，取值范围：0~3000,000，0 表示恢复默认设置

byRes

保留，置为 0

See Also

[NET_DVR_GetSDKLocalCfg](#)、[NET_DVR_SetSDKLocalCfg](#)

7.46 NET_DVR_LOCAL_PROTECT_KEY_CFG:密钥配置

```
struct{
    BYTE     byProtectKey[128];
    BYTE     byRes[128];
}NET_DVR_LOCAL_PROTECT_KEY_CFG,*LPNET_DVR_LOCAL_PROTECT_KEY_CFG;
```

Members

byProtectKey

密钥，默认设置为 0

byRes

保留，置为 0

7.47 NET_DVR_LOCAL_TALK_MODE_CFG:对讲模式配置

```
struct{
    BYTE    byEnableAbilityParse;
    BYTE    byRes[127];
}NET_DVR_LOCAL_TALK_MODE_CFG,*LPNET_DVR_LOCAL_TALK_MODE_CFG;
```

Members

byTalkMode

对讲模式：0- 使用对讲库（默认），1- 使用 windows api 模式

byRes

保留，置为 0

Remarks

V4.2.2.5 及以前版本 SDK 均采用 windows API 实现相关功能。之后版本默认使用语音对讲库的方式，语音对讲库模式下必须加载 AudioIntercom.dll 和 OpenAL32.dll。

See Also

[NET_DVR_GetSDKLocalCfg](#)、[NET_DVR_SetSDKLocalCfg](#)

7.48 NET_DVR_LOCAL_TCP_PORT_BIND_CFG:本地 TCP 端口绑定配置

```
struct{
    WORD    wLocalBindTcpMinPort;
    WORD    wLocalBindTcpMaxPort;
    BYTE    byRes[60];
}NET_DVR_LOCAL_TCP_PORT_BIND_CFG,*LPNET_DVR_LOCAL_TCP_PORT_BIND_CFG;
```

Members

wLocalBindTcpMinPort

本地绑定 TCP 最小端口

wLocalBindTcpMaxPort

本地绑定 TCP 最大端口

byRes

保留，置为 0

Remarks

端口绑定的策略是：给一个端口段，可以保证使用的端口都是在这个段里（多播除外），但不能保证每一个段内的端口都用到，因为是循环利用的；端口池中取出的端口会去尝试绑定，如果被占用了，将取下一个，如果段内每一个都绑定不了，则连接操作返回失败。建议最好不要设置系统预留的端口（1-1024），比如 80 等。

设置的最大端口应该大于等于最小端口，[0, 0]表示清除绑定，[0, 非 0]将设置失败，因为 0 不能进行绑定。

See Also

[NET_DVR_GetSDKLocalCfg](#)、[NET_DVR_SetSDKLocalCfg](#)

7.49 NET_DVR_LOCAL_UDP_PORT_BIND_CFG:本地 UDP 端口绑定配置

```
struct{
    WORD    wLocalBindUdpMinPort;
    WORD    wLocalBindUdpMaxPort;
    BYTE    byRes[60];
}NET_DVR_LOCAL_UDP_PORT_BIND_CFG,*LPNET_DVR_LOCAL_UDP_PORT_BIND_CFG;
```

Members

wLocalBindUdpMinPort

本地绑定 UDP 最小端口

wLocalBindUdpMaxPort

本地绑定 UDP 最大端口

byRes

保留，置为 0

Remarks

端口绑定的策略是：给一个端口段，可以保证使用的端口都是在这个段里（多播除外），但不能保证每一个段内的端口都用到，因为是循环利用的；端口池中取出的端口会去尝试绑定，如果被占用了，将取下一个，如果段内每一个都绑定不了，则连接操作返回失败。建议最好不要设置系统预留的端口（1-1024），比如 80 等。

设置的最大端口应该大于等于最小端口，[0,0]表示清除绑定，[0, 非 0]将设置失败，因为 0 不能进行绑定。

See Also

[NET_DVR_GetSDKLocalCfg](#)、[NET_DVR_SetSDKLocalCfg](#)

7.50 NET_DVR_LOG_V30:日志信息

```
struct{
    NET\_DVR\_TIME    strLogTime;
    DWORD            dwMajorType;
    DWORD            dwMinorType;
    BYTE             sPanelUser[MAX_NAMELEN];
    BYTE             sNetUser[MAX_NAMELEN];
    NET\_DVR\_IPADDR   struRemoteHostAddr;
    DWORD            dwParaType;
    DWORD            dwChannel;
    DWORD            dwDiskNumber;
    DWORD            dwAlarmInPort;
    DWORD            dwAlarmOutPort;
    DWORD            dwInfoLen;
    char             sInfo[LOG_INFO_LEN];
}NET_DVR_LOG_V30,*LPNET_DVR_LOG_V30;
```

Members

strLogTime

日志时间

dwMajorType

报警主类型，定义请参见[日志主类型列表](#)

dwMinorType

报警次类型，根据不同的主类型的次类型定义请参见[日志次类型列表](#)

sPanelUser

操作面板的用户名

sNetUser

网络操作的用户名

struRemoteHostAddr

远程主机地址

dwParaType

当日志的主类型为 MAJOR_OPERATION=03（操作），且次类型为 MINOR_LOCAL_CFG_PARM=0x52（本地配置参数）或 MINOR_REMOTE_GET_PARM=0x76（远程获得参数）或 MINOR_REMOTE_CFG_PARM=0x77（远程配置参数）时，该参数类型有效，其含义如下：

宏定义	宏定义值	含义
PARAM_VIDEOOUT	0x1	视频输出结构配置
PARAM_IMAGE	0x2	图像参数结构配置
PARAM_ENCODE	0x4	压缩参数结构配置
PARAM_NETWORK	0x8	网络参数结构配置
PARAM_ALARM	0x10	报警参数结构配置
PARAM_EXCEPTION	0x20	异常参数结构配置
PARAM_DECODER	0x40	解码器参数结构配置
PARAM_RS232	0x80	RS232 参数结构配置
PARAM_PREVIEW	0x100	本地预览参数结构配置
PARAM_SECURITY	0x200	用户权限参数结构配置
PARAM_DATETIME	0x400	本地系统配置
PARAM_FRAMETYPE	0x800	帧信息参数结构配置
PARAM_CODESPLITTER	0x2000	码分器参数
PARAM_RS485	0x2001	RS485 配置信息
PARAM_DEVICE	0x2002	设备配置信息
PARAM_HARDDISK	0x2003	硬盘配置信息
PARAM_AUTOBOOT	0x2004	自动重启配置信息
PARAM_HOLIDAY	0x2005	节假日配置信息
PARAM_IPC	0x2006	IP 通道配置

dwChannel

通道号

dwDiskNumber

硬盘号

dwAlarmInPort

报警输入端口

dwAlarmOutPort

报警输出端口

dwInfoLen

日志附加信息长度

sInfo

日志附加信息

See Also

[NET_DVR_FindNextLog_V30](#)

7.51 NET_DVR_MANAGE_UNIT_DEVICEID: 管理机编号参数结构体

```
struct{
    WORD    wPeriod;
    WORD    wDevIndex;
    BYTE    byRes[124];
}NET_DVR_MANAGE_UNIT_DEVICEID,*LPNET_DVR_MANAGE_UNIT_DEVICEID;
```

Members

wPeriod

期号，取值范围：[0,9]

wDevIndex

管理机序号，每一层中门口机的序号唯一，从 0 开始

byRes

保留，置为 0

See Also

[NET_DVR_VIDEO_INTERCOM_UNIT_DEVICEID_UNION](#)

7.52 NET_DVR_MANAGE_UNIT_OPERATION_TIME_CFG: 管理机操作时间配置结构体

```
struct{
    DWORD    dwMaxMonitoringTime;
    DWORD    dwMaxRingTime;
    DWORD    dwMaxTalkTime;
    BYTE     byRes[116];
}NET_DVR_MANAGE_UNIT_OPERATION_TIME_CFG,*LPNET_DVR_MANAGE_UNIT_OPERATION_TIME_CFG;
```

Members

dwMaxMonitoringTime

最大监视时间，范围：[10,60]，单位：秒

dwMaxRingTime

最大振铃时间，范围：[15,60]，单位：秒

dwMaxTalkTime

最大通话时间，单位：秒

byRes

保留，置为 0

See Also

[NET_DVR_VIDEO_INTERCOM_OPERATION_TIME_UNION](#)

7.53 NET_DVR_MANAGE_UNIT_RELATEDDEV: 管理机关联网络设备参数结构体

```
struct{
    NET\_DVR\_IPADDR    struSIPServer;
    BYTE              byRes[880];
}NET_DVR_MANAGE_UNIT_RELATEDDEV,*LPNET_DVR_MANAGE_UNIT_RELATEDDEV;
```

Members

struSIPServer

SIP 服务器 IP 地址

byRes

保留，置为 0

See Also

[NET_DVR_VIDEO_INTERCOM_UNIT_RELATEDDEV_UNION](#)

7.54 NET_DVR_MOTION:移动侦测参数结构体

```
struct{
    BYTE              byMotionScope[18][22];
    BYTE              byMotionSensitive;
    BYTE              byEnableHandleMotion;
    BYTE              reservedData[2];
    NET\_DVR\_HANDLEEXCEPTION strMotionHandleType;
    NET\_DVR\_SCHEDTIME    struAlarmTime[MAX_DAYS][MAX_TIMESEGMENT];
    BYTE              byRelRecordChan[MAX_CHANNUM];
}NET_DVR_MOTION,*LPNET_DVR_MOTION;
```

Members

byMotionScope

移动侦测区域，共有 22*18 个小宏块，取值为 1 表示该宏块设定为移动侦测区域，0 表示不设定为移动侦测区域

byMotionSensitive

移动侦测灵敏度，取值范围[0,5]，若取值为 0xff 表示关闭，值越大越灵敏

byEnableHandleMotion

是否处理移动侦测：0-不处理，1-处理

reservedData

保留，置为 0

strMotionHandleType

处理方式

struAlarmTime

布防时间

byRelRecordChan

报警触发的录象通道，为 1 表示触发该通道

Remarks

整个图像的区域大小为 704*576

See Also

[NET_DVR_PICCFG](#)

7.55 NET_DVR_NETAPPCFG: 网络应用参数（NTP/DDNS）结构体

```
struct{
    DWORD                dwSize;
    char                 sDNSIp[16];
    NET\_DVR\_NTPPARA      struNtpClientParam;
    NET\_DVR\_DDNSPARA     struDDNSClientParam;
    BYTE                res[464];
}NET_DVR_NETAPPCFG,*LPNET_DVR_NETAPPCFG;
```

Members

dwSize

结构体大小

sDNSIp

DNS 服务器地址

struNtpClientParam

NTP 参数

struDDNSClientParam

DDNS 参数

res

保留，置为 0

See Also

[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)

7.56 NET_DVR_NETCFG_V30:网络配置

```
struct{
    DWORD                dwSize;
```

```

NET\_DVR\_ETHERNET\_V30    struEtherNet[MAX_ETHERNET];
NET\_DVR\_IPADDR          struRes1[2];
NET\_DVR\_IPADDR          struAlarmHostIpAddr;
WORD                      wRes2[2];
WORD                      wAlarmHostIpPort;
BYTE                      byUseDhcp;
BYTE                      byRes3;
NET\_DVR\_IPADDR          struDnsServer1IpAddr;
NET\_DVR\_IPADDR          struDnsServer2IpAddr;
BYTE                      byIpResolver[MAX_DOMAIN_NAME];
WORD                      wIpResolverPort;
WORD                      wHttpPortNo;
NET\_DVR\_IPADDR          struMulticastIpAddr;
NET\_DVR\_IPADDR          struGatewayIpAddr;
NET\_DVR\_PPPOECFG        struPPPoE;
BYTE                      byRes[64];
}NET_DVR_NETCFG_V30,*LPNET_DVR_NETCFG_V30;

```

Members

dwSize

结构体大小

struEtherNet

以太网口

struRes1

保留，置为 0

struAlarmHostIpAddr

报警主机 IP 地址

wRes2

保留，置为 0

wAlarmHostIpPort

报警主机端口号

byUseDhcp

是否启用 DHCP: 0xff-无效; 0-不启用; 1-启用

byRes3

保留，置为 0

struDnsServer1IpAddr

域名服务器 1 的 IP 地址

struDnsServer2IpAddr

域名服务器 2 的 IP 地址

byIpResolver

IP 解析服务器域名或 IP 地址（8000 设备不支持域名）

wIpResolverPort

IP 解析服务器端口号

wHttpPortNo

HTTP 端口号

`struMulticastIpAddr`

多播组地址

`struGatewayIpAddr`

网关地址

`struPPPoE`

PPPoE 参数

`byRes`

保留，置为 0

Remarks

8000 等 3.0 协议以下的设备，参数 `byUseDhcp` 为 0xff-无效，将其 IP 地址填成空，设备会自动去获取 DHCP。

See Also

[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)

7.57 **NET_DVR_NOTICE_DATA**:公告数据下发参数

```
struct{
    DWORD                dwSize;
    NET\_DVR\_TIME\_EX      struTime;
    BYTE                 byNoticeNumber[MAX\_NOTICE\_NUMBER\_LEN];
    BYTE                 byNoticeTheme[MAX\_NOTICE\_THEME\_LEN];
    BYTE                 byNoticeDetail[MAX\_NOTICE\_DETAIL\_LEN];
    BYTE                 byLevel;
    BYTE                 byPicNum;
    BYTE                 byRes1[2];
    NET\_DVR\_NOTICE\_PIC   struNoticePic[MAX\_NOTICE\_PIC\_NUM];
    BYTE                 byRes2[128];
}NET_DVR_NOTICE_DATA, *LPNET_DVR_NOTICE_DATA;
```

Members

dwSize

结构体大小

struTime

公告时间

byNoticeNumber

公告编号

byNoticeTheme

公告主题

byNoticeDetail

公告详情

byLevel

公告等级：1- 广告类信息，2- 物业信息，3- 报警类信息，4- 通知类信息

byPicNum

公告图片数量

byRes1

保留，置为 0

struNoticePic

公告图片信息

byRes2

保留，置为 0

Remarks

- 公告数据下发参数能力，对应 IP 可视对讲主机能力集（接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：[IP_VIEW_DEV_ABILITY](#)）中节点<NoticeData>。

See Also

[NET_DVR_RemoteControl](#)

7.58 NET_DVR_NOTICEDATA_RECEIPT_INFO:公告信息阅读回执信息

```
struct{
    BYTE    byNoticeNumber[MAX_NOTICE_NUMBER_LEN];
    BYTE    byRes[224];
}NET_DVR_NOTICEDATA_RECEIPT_INFO, *LPNET_DVR_NOTICEDATA_RECEIPT_INFO;
```

Members

byNoticeNumber

公告编号

byRes

保留，置为 0

See Also

[NET_DVR_VIDEO_INTERCOM_EVENT_INFO_UINON](#)

7.59 NET_DVR_NOTICE_PIC:公告图片信息

```
struct{
    BYTE    *pPicData;
    DWORD   dwPicDataLen;
    BYTE    byRes[32];
}NET_DVR_NOTICE_PIC, *LPNET_DVR_NOTICE_PIC;
```

Members

pPicData

缓冲区指针，保存图片数据（单张图片长度最大 512KB，默认 jpeg 格式）

dwPicDataLen

图片数据长度

byRes

保留，置为 0

See Also

[NET_DVR_NOTICE_DATA](#)

7.60 NET_DVR_NTTPARA:网络应用参数(NTP)

```
struct{
    BYTE          sNTPServer[64];
    WORD          wInterval;
    BYTE          byEnableNTP;
    signed char    cTimeDifferenceH;
    signed char    cTimeDifferenceM;
    BYTE          res1;
    WORD          wNtpPort;
    BYTE          res2[8];
}NET_DVR_NTTPARA,*LPNET_DVR_NTTPARA;
```

Members

sNTPServer

NTP 服务器域名或者 IP 地址

wInterval

校时间隔时间，以分钟或小时为单位（通过能力集 [NET_DVR_GetDeviceAbility](#) 获取，对应网络应用参数能力集：DEVICE_NETAPP_ABILITY）

byEnableNTP

NTP 校时是否启用：0—否，1—是

cTimeDifferenceH

与国际标准时间的时差（小时），-12 ... +13

cTimeDifferenceM

与国际标准时间的时差（分钟），0, 30, 45

res1

保留，置为 0

wNtpPort

NTP 服务器端口，设备默认为 123

res2

保留，置为 0

See Also

[NET_DVR_GetDVRConfig](#)

7.61 NET_DVR_OUTDOOR_FENCE_DEVICEID:围墙机编号参数结构体

```
struct{
    WORD          wPeriod;
    WORD          wDevIndex;
    BYTE          byRes[124];
}NET_DVR_OUTDOOR_FENCE_DEVICEID,*LPNET_DVR_OUTDOOR_FENCE_DEVICEID;
```

Members

wPeriod

期号，取值范围：[0,9]

wDevIndex

围墙机序号，每一层中门口机的序号唯一，从 0 开始

byRes

保留，置为 0

See Also

[NET_DVR_VIDEO_INTERCOM_UNIT_DEVICEID_UNION](#)

7.62 NET_DVR_OUTDOOR_UNIT_DEVICEID: 门口机编号参数结构体

```
struct{
    WORD    wPeriod;
    WORD    wBuildingNumber;
    WORD    wUnitNumber;
    WORD    wFloorNumber;
    WORD    wDevIndex;
    BYTE    byRes[118];
}NET_DVR_OUTDOOR_UNIT_DEVICEID,*LPNET_DVR_OUTDOOR_UNIT_DEVICEID;
```

Members

wPeriod

期号，取值范围：[0,9]

wBuildingNumber

楼号

wUnitNumber

单元号

wFloorNumber

层号

wDevIndex

门口机序号，每一层中门口机的序号唯一，从 0 开始

byRes

保留，置为 0

See Also

[NET_DVR_VIDEO_INTERCOM_UNIT_DEVICEID_UNION](#)

7.63 NET_DVR_OUTDOOR_UNIT_OPERATION_TIME_CFG: 门口机/围墙机/别墅门口机/二次确认机操作时间配置结构体

```
struct{
    DWORD    dwMaxMessageTime;
    DWORD    dwMaxTalkTime;
    BYTE     byRes[120];
}NET_DVR_OUTDOOR_UNIT_OPERATION_TIME_CFG,*LPNET_DVR_OUTDOOR_UNIT_OPERATION_TIME_CFG;
```

Members

dwMaxMessageTime

最大留言时间，范围：[30,60]，单位：秒

dwMaxTalkTime

最大通话时间，范围：[90,120]，单位：秒

byRes

保留，置为 0

See Also[NET_DVR_VIDEO_INTERCOM_OPERATION_TIME_UNION](#)

7.64 **NET_DVR_OUTDOOR_UNIT_RELATEDDEV**: 门口机/围墙机/别墅门口机关联网络设备参数结构体

```
struct{
    NET\_DVR\_IPADDR    struMainOutdoorUnit;
    NET\_DVR\_IPADDR    struManageUnit;
    NET\_DVR\_IPADDR    struSIPServer;
    BYTE              byRes[592];
}NET_DVR_OUTDOOR_UNIT_RELATEDDEV,*LPNET_DVR_OUTDOOR_UNIT_RELATEDDEV;
```

Members

struMainOutdoorUnit

主门口机 IP 地址，副门口机时有效

struManageUnit

管理机 IP 地址

struSIPServer

SIP 服务器 IP 地址，副门口机时无效

byRes

保留，置为 0

See Also[NET_DVR_VIDEO_INTERCOM_UNIT_RELATEDDEV_UNION](#)

7.65 **NET_DVR_PICCFG**:通道图像结构体

```
struct{
    DWORD              dwSize;
    BYTE               sChanName[NAME_LEN];
    DWORD              dwVideoFormat;
    BYTE               byRes[4];
    DWORD              dwShowChanName;
    WORD               wShowNameTopLeftX;
```

```

WORD                                wShowNameTopLeftY;
NET\_DVR\_VILOST                    struVILost;
NET\_DVR\_MOTION                    struMotion;
NET\_DVR\_HIDEALARM                struHideAlarm;
DWORD                                dwEnableHide;
WORD                                wHideAreaTopLeftX;
WORD                                wHideAreaTopLeftY;
WORD                                wHideAreaWidth;
WORD                                wHideAreaHeight;
DWORD                                dwShowOsd;
WORD                                wOSDTopLeftX;
WORD                                wOSDTopLeftY;
BYTE                                byOSDType;
BYTE                                byDispWeek;
BYTE                                byOSDAttrib;
char                                reservedData2;
}NET_DVR_PICCFG,*LPNET_DVR_PICCFG;

```

Members

dwSize

结构体大小

sChanName

通道名称

dwVideoFormat

视频制式：1-NTSC；2-PAL

byRes

保留

dwShowChanName

预览的图象上是否显示通道名称：0-不显示，1-显示（区域大小 704*576）

wShowNameTopLeftX

通道名称显示位置的 x 坐标

wShowNameTopLeftY

通道名称显示位置的 y 坐标

struVILost

信号丢失报警参数

struMotion

移动侦测报警参数

struHideAlarm

遮挡报警参数

dwEnableHide

是否启动隐私遮蔽：0-否，1-是

wHideAreaTopLeftX

隐私遮蔽区域的 x 坐标

wHideAreaTopLeftY

隐私遮蔽区域的 y 坐标

wHideAreaWidth

隐私遮蔽区域的宽

wHideAreaHeight

隐私遮蔽区域的高

dwShowOsd

预览的图象上是否显示 OSD: 0-不显示, 1-显示 (区域大小 704*576)

wOSDTopLeftX

OSD 的 x 坐标

wOSDTopLeftY

OSD 的 y 坐标

byOSDType

OSD 类型(年月日格式):

0—XXXX-XX-XX 年月日

1—XX-XX-XXXX 月日年

2—XXXX 年 XX 月 XX 日

3—XX 月 XX 日 XXXX 年

4—XX-XX-XXXX 日月年

5—XX 日 XX 月 XXXX 年

byDispWeek

是否显示星期: 0-不显示, 1-显示

byOSDAttrib

OSD 属性 (透明/闪烁):

1—透明, 闪烁

2—透明, 不闪烁

3—闪烁, 不透明

4—不透明, 不闪烁

reservedData2

保留, 置为 0

See Also

[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)

7.66 NET_DVR_PICTURE_NAME_EX: 图片命名规则扩展

struct{

BYTE byItemOrder[PICNAME_MAXITEM];

BYTE byDelimiter;

BYTE byPicNamePrefix[NAME_LEN];

}NET_DVR_PICTURE_NAME_EX, *LPNET_DVR_PICTURE_NAME_EX;

Members

byItemOrder

按数组定义文件命名的规则, 每个元素取值如下所示:

宏定义	宏定义值	含义
-----	------	----

PICNAME_ITEM_PARK_DEV_IP	1	设备 IP 地址
PICNAME_ITEM_PARK_PLATE_NO	2	车牌号码
PICNAME_ITEM_PARK_TIME	3	时间
PICNAME_ITEM_PARK_INDEX	4	车位编号
PICNAME_ITEM_PARK_STATUS	5	车位状态
PICNAME_ITEM_BUILDING_NUMBER	6	栋号单元号
PICNAME_ITEM_OUTDOOR_UNIT_ID	7	门口机编号
PICNAME_ITEM_UNLOCK_TYPE	8	开锁方式
PICNAME_ITEM_DEVICE_NAME	9	设备名称
PICNAME_ITEM_PERIOD_NO	10	期号
PICNAME_ITEM_DEV_INDEX	11	设备编号

byDelimiter

分隔符，一般为 '_'

byPicNamePrefix

图片名自定义前缀，格式为：自定义前缀_设备序列号_时间_类型.jpg；前缀默认为空，支持用户输入，包括数字、符号（除特殊符合，如：不能含有字符 / \ : * ? ' " < > | %）、英文等，最大长度 32 个字符（16 个汉字）

See Also

[NET_DVR_FTPCFG_V40](#)

7.67 NET_DVR_PPPCFG_V30:PPP 参数结构体

```
struct{
    NET\_DVR\_IPADDR    struRemoteIP;
    NET\_DVR\_IPADDR    struLocalIP;
    char              sLocalIPMask[16];
    BYTE              sUsername[NAME_LEN];
    BYTE              sPassword[PASSWD_LEN];
    BYTE              byPPPMODE;
    BYTE              byRedial;
    BYTE              byRedialMode;
    BYTE              byDataEncrypt;
    DWORD             dwMTU;
    char              sTelephoneNumber[PHONENUMBER_LEN];
}NET_DVR_PPPCFG_V30, *LPNET_DVR_PPPCFG_V30;
```

Members

struRemoteIP

远端 IP 地址

struLocalIP

本地 IP 地址

sLocalIPMask

本地 IP 地址掩码

sUsername

用户名

sPassword

密码

byPPPMMode

PPP 模式：0- 主动，1- 被动

byRedial

是否回拨：0-否，1-是

byRedialMode

回拨模式：0-由拨入者指定，1-预置回拨号码

byDataEncrypt

数据加密：0-否，1-是

dwMTU

MTU

sTelephoneNumber

电话号码

See Also

[NET_DVR_RS232CFG_V30](#)

7.68 NET_DVR_PPPOECFG:PPPoE 配置结构体

```
struct{
    DWORD          dwPPPOE;
    BYTE           sPPPoEUser[NAME_LEN];
    char           sPPPoEPassword[PASSWD_LEN];
    NET\_DVR\_IPADDR struPPPoEIP;
}NET_DVR_PPPOECFG, *LPNET_DVR_PPPOECFG;
```

Members

dwPPPOE

是否启用 PPPoE：0-不启用，1-启用

sPPPoEUser

PPPoE 用户名

sPPPoEPassword

PPPoE 密码

struPPPoEIP

PPPoE IP 地址

See Also

[NET_DVR_NETCFG_V30](#)

7.69 NET_DVR_PPPOECFG: 预览参数结构体

```
struct{
    LONG        IChannel;
    DWORD       dwStreamType;
    DWORD       dwLinkMode;
    HWND        hPlayWnd;
    BOOL        bBlocked;
    BOOL        bPassbackRecord;
    BYTE        byPreviewMode;
    BYTE        byStreamID[STREAM_ID_LEN];
    BYTE        byProtoType;
    BYTE        byRes1[2];
    DWORD       dwDisplayBufNum;
    BYTE        byRes[216];
}NET_DVR_PREVIEWINFO, *LPNET_DVR_PREVIEWINFO;
```

Members

IChannel

通道号，目前设备模拟通道号从 1 开始，数字通道的起始通道号通过 NET_DVR_GetDVRConfig（配置命令 NET_DVR_GET_IPPARACFG_V40）获取（dwStartDChan）。

dwStreamType

码流类型：0-主码流，1-子码流，2-码流 3，3-虚拟码流，以此类推

dwLinkMode

连接方式：0- TCP 方式，1- UDP 方式，2- 多播方式，3- RTP 方式，4- RTP/RTSP，5- RTP/HTTP，6- HRUDP（可靠传输）

hPlayWnd

播放窗口的句柄，为 NULL 表示不解码显示。

bBlocked

0- 非阻塞取流，1- 阻塞取流。如果阻塞取流，SDK 内部 connect 失败将会有 5s 的超时才能够返回，不适合于轮询取流操作。

bPassbackRecord

0-不启用录像回传，1-启用录像回传。ANR 断网补录功能，客户端和设备之间网络异常恢复之后自动将前端数据同步过来，需要设备支持。

byPreviewMode

预览模式：0- 正常预览，1- 延迟预览

byStreamID

流 ID，为字母、数字和"_"的组合，IChannel 为 0xffffffff 时启用此参数

byProtoType

应用层取流协议：0- 私有协议，1- RTSP 协议。主子码流支持的取流协议通过登录返回结构参数 NET_DVR_DEVICEINFO_V30 的 byMainProto、bySubProto 值得知。设备同时支持私协议和 RTSP 协议时，该参数才有效，默认使用私有协议，可选 RTSP 协议。

byRes1

保留，置为 0

dwDisplayBufNum

播放库播放缓冲区最大缓冲帧数，取值范围：1~50，置 0 时默认为 1。设置显示缓冲需要在播放库调用 PlayM4_Play 之前调用，该参数替换原先 NET_DVR_SetPlayerBufNumber 接口

byRes

保留，置为 0

Remarks

- 该结构体中可以设置当前预览操作是否阻塞（通过 bBlocked 参数设置）。若设为不阻塞，表示发起与设备的连接就认为连接成功，如果发生码流接收失败、播放失败等情况以预览异常的方式通知上层。在循环播放的时候可以减短停顿的时间，与 NET_DVR_RealPlay 处理一致。若设为阻塞，表示直到播放操作完成才返回成功与否。
- dwStreamType、dwLinkMode、bPassbackRecord、byPreviewMode、byStreamID 这些参数的取值需要设备支持。
- NET_DVR_RealPlay_V40 支持多播方式预览（dwLinkMode 设为 2），不需要传多播组地址，底层自动从设备获取已配置的多播组地址（NET_DVR_NETCFG_V30->struMulticastIpAddr）并以该多播组地址实现多播。
- Linux 64 位系统不支持软解码功能，因此需要将窗口句柄传 NULL，设置回调函数，只取流不解码显示。

See Also

[NET_DVR_RealPlay_V40](#)

7.70 NET_DVR_PRIVILEGE_PASSWORD_CFG: 权限密码配置

struct{

```
DWORD    dwSize;
BYTE     byPwdType;
BYTE     byRes1[3];
BYTE     byOldPassword[PASSWD_LEN];
BYTE     byNewPassword[PASSWD_LEN];
BYTE     byRes2[128];
```

}NET_DVR_PRIVILEGE_PASSWORD_CFG, *LPNET_DVR_PRIVILEGE_PASSWORD_CFG;

Members

dwSize

结构体大小

byPwdType

密码类型，具体定义如下：

```
enum _PRIVILEGE_PASSWORD_TYPE_ENUM_{
    ENUM_PRIVILEGE_PASSWORD_ENGINEERING    = 1,
    ENUM_PRIVILEGE_PASSWORD_SETUPALARM      = 2,
    ENUM_PRIVILEGE_PASSWORD_HOUSEHOLDER_UNLOCK = 3,
    ENUM_PRIVILEGE_PASSWORD_ANTI_HIJACKING  = 4
}PRIVILEGE_PASSWORD_TYPE_ENUM
```

ENUM_PRIVILEGE_PASSWORD_ENGINEERING

工程密码

ENUM_PRIVILEGE_PASSWORD_SETUPALARM

布/撤防密码

ENUM_PRIVILEGE_PASSWORD_HOUSEHOLDER_UNLOCK

户主开锁密码
ENUM_PRIVILEGE_PASSWORD_ANTI_HIJACKING
 防劫持密码

byRes1

保留，置为 0

byOldPassword

旧密码

byNewPassword

新密码

byRes2

保留，置为 0

Remarks

- 权限密码配置能力，对应 IP 可视对讲主机能力集（接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：[IP_VIEW_DEV_ABILITY](#)）中节点<PrivilegePwd>。

See Also

[NET_DVR_SetDVRConfig](#)

7.71 **NET_DVR_PU_STREAM_URL:URL** 取流配置

```
struct{
    BYTE    byEnable;
    BYTE    strURL[240];
    BYTE    byTransPortocol;
    WORD    wIPID;
    BYTE    byChannel;
    BYTE    byRes[7];
}NET_DVR_PU_STREAM_URL,*LPNET_DVR_PU_STREAM_URL;
```

Members

byEnable

是否启用：0- 禁用，1- 启用

strURL

取流 URL 路径

byTransPortocol

传输协议类型：0-TCP，1-UDP

wIPID

设备 ID 号，wIPID = iDevInfoIndex + iGroupNO*64 +1

byChannel

设备通道号

byRes

保留，置为 0

See Also

[NET_DVR_STREAM_TYPE_UNION](#)

7.72 NET_DVR_PU_STREAM_URL_CFG:URL 取流路径构成参数

```
struct{
    BYTE    byEnable;
    BYTE    byRes[3];
    BYTE    byStreamMediaIP[64];
    WORD    wStreamMediaPort;
    BYTE    byTransmitType;
    BYTE    byRes1[33];
    BYTE    byDevIP[64];
    WORD    wDevPort;
    BYTE    byChannel;
    BYTE    byTransMode;
    BYTE    byProType;
    BYTE    byTransProtocol;
    BYTE    byRes3[2];
    BYTE    sUserName[NAME_LEN];
    BYTE    sPassWord[PASSWD_LEN];
    BYTE    byRes2[28];
}NET_DVR_PU_STREAM_URL_CFG,*LPNET_DVR_PU_STREAM_URL_CFG;
```

Members

byEnable

是否启用：0- 禁用，1- 启用

byRes

保留，置为 0

byStreamMediaIP

流媒体 IP 地址

wStreamMediaPort

流媒体端口

byTransmitType

流媒体传输协议：0- TCP，1- UDP，2-RTSP

byRes1

保留，置为 0

byDevIP

设备 IP 地址

wDevPort

设备端口号

byChannel

设备通道号

byTransMode

传输模式：0- 主码流，1- 子码流

byProType

厂家协议类型：0- 私有协议，1- 大华，2- 汉邦，3- 郎驰，4- 蓝色星际

byTransProtocol

传输协议类型：0- TCP，1- UDP，2- 多播方式，3- RTP

byRes3

保留，置为 0

sUserName

设备登陆用户名

sPassword

设备登陆密码

byRes

保留，置为 0

See Also

[NET_DVR_STREAM_TYPE_UNION](#)

7.73 **NET_DVR_RECTCFG**:窗口设置结构体

```
struct{
    WORD    wXCoordinate;
    WORD    wYCoordinate;
    WORD    wWidth;
    WORD    wHeight;
}NET_DVR_RECTCFG, *LPNET_DVR_RECTCFG;
```

Members

wXCoordinate

矩形左上角起始点 X 坐标

wYCoordinate

矩形左上角 Y 坐标

wWidth

矩形宽度

wHeight

矩形高度

See Also

[NET_DVR_AREA_ZOOM_CFG](#)

7.74 **NET_DVR_RECTCFG_EX**:矩形窗口参数

```
struct{
    DWORD    dwXCoordinate;
    DWORD    dwYCoordinate;
    DWORD    dwWidth;
    DWORD    dwHeight;
    BYTE     byRes[4];
}NET_DVR_RECTCFG_EX, *LPNET_DVR_RECTCFG_EX;
```

Members**dwXCoordinate**

矩形左上角起始点 X 坐标，对于电视墙窗口该值为基准 X 坐标的倍数

dwYCoordinate

矩形左上角 Y 坐标，对于电视墙窗口该值为基准 Y 坐标的倍数

dwWidth

矩形宽度，对于电视墙窗口该值为基准 X 坐标的倍数

dwHeight

矩形高度，对于电视墙窗口该值为基准 Y 坐标的倍数

byRes

保留，置为 0

See Also**7.75 NET_DVR_RS232CFG_V30:RS232 串口参数结构体**

```

struct{
    DWORD                dwSize;
    NET\_DVR\_SINGLE\_RS232 struRs232[MAX_SERIAL_PORT];
    NET\_DVR\_PPPCFG\_V30   struPPPConfig;
}NET_DVR_RS232CFG_V30, *LPNET_DVR_RS232CFG_V30;

```

Members**dwSize**

结构体大小

struRs232

只有一个 232 串口，采用第一个

struPPPConfig

PPP 参数

See Also[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)**7.76 NET_DVR_SCHEDULETIME:起止时间段参数**

```

struct{
    BYTE    byStartHour;
    BYTE    byStartMin;
    BYTE    byStopHour;
    BYTE    byStopMin;
}NET_DVR_SCHEDULETIME, *LPNET_DVR_SCHEDULETIME;

```

Members**byStartHour**

开始时间：时

byStartMin

开始时间：分

byStopHour

结束时间：时

byStopMin

结束时间：分

See Also

[NET_DVR_VILOS](#)、[NET_DVR_MOTION](#)、[NET_DVR_HIDEALARM](#)

7.77 **NET_DVR_SDKSTATE**:SDK 状态信息

```
struct{
    DWORD      dwTotalLoginNum;
    DWORD      dwTotalRealPlayNum;
    DWORD      dwTotalPlayBackNum;
    DWORD      dwTotalAlarmChanNum;
    DWORD      dwTotalFormatNum;
    DWORD      dwTotalFileSearchNum;
    DWORD      dwTotalLogSearchNum;
    DWORD      dwTotalSerialNum;
    DWORD      dwTotalUpgradeNum;
    DWORD      dwTotalVoiceComNum;
    DWORD      dwTotalBroadCastNum;
    DWORD      dwRes[10];
}NET_DVR_SDKSTATE,*LPNET_DVR_SDKSTATE;
```

Members

dwTotalLoginNum

当前注册的用户数

dwTotalRealPlayNum

当前实时预览的路数

dwTotalPlayBackNum

当前回放或下载的路数

dwTotalAlarmChanNum

当前建立报警通道的路数

dwTotalFormatNum

当前硬盘格式化的路数

dwTotalFileSearchNum

当前文件搜索的路数

dwTotalLogSearchNum

当前日志搜索的路数

dwTotalSerialNum

当前建立透明通道的路数

dwTotalUpgradeNum

当前升级的路数

dwTotalVoiceComNum

当前语音转发的路数

dwTotalBroadCastNum

当前语音广播的路数

dwRes

保留，置为 0

See Also

[NET_DVR_GetSDKState](#)

7.78 NET_DVR_SDKABL:SDK 功能信息

```
struct{
    DWORD    dwMaxLoginNum;
    DWORD    dwMaxRealPlayNum;
    DWORD    dwMaxPlayBackNum;
    DWORD    dwMaxAlarmChanNum;
    DWORD    dwMaxFormatNum;
    DWORD    dwMaxFileSearchNum;
    DWORD    dwMaxLogSearchNum;
    DWORD    dwMaxSerialNum;
    DWORD    dwMaxUpgradeNum;
    DWORD    dwMaxVoiceComNum;
    DWORD    dwMaxBroadCastNum;
    DWORD    dwRes[10];
}NET_DVR_SDKABL,*LPNET_DVR_SDKABL;
```

Members

dwMaxLoginNum

最大注册用户数

dwMaxRealPlayNum

最大实时预览的路数

dwMaxPlayBackNum

最大回放或下载的路数

dwMaxAlarmChanNum

最大建立报警通道的路数

dwMaxFormatNum

最大硬盘格式化的路数

dwMaxFileSearchNum

最大文件搜索的路数

dwMaxLogSearchNum

最大日志搜索的路数

dwMaxSerialNum

最大建立透明通道的路数

dwMaxUpgradeNum

最大升级的路数

dwMaxVoiceComNum

最大语音转发的路数

dwMaxBroadCastNum

最大语音广播的路数

dwRes

保留，置为 0

See Also

[NET_DVR_GetSDKAbility](#)

7.79 NET_DVR_SERIAL_CONTROL: 串口信息结构体

```
struct{
    DWORD    dwSize;
    BYTE     bySerialNum;
    BYTE     byRes1[3];
    BYTE     bySerial[32];
    BYTE     byRes2[32];
}NET_DVR_SERIAL_CONTROL, *LPNET_DVR_SERIAL_CONTROL;
```

Members

dwSize

结构体大小

bySerialNum

串口个数

byRes1

保留

bySerial

有效串口序号，数组的前 bySerialNum 个数据有效，bySerial[0]=1 表示 COM1

byRes2

保留

See Also

[NET_DVR_GetDVRConfig](#)

7.80 NET_DVR_SETUPALARM_PARAM: 报警布防参数

```
struct{
    DWORD    dwSize;
    BYTE     byLevel;
    BYTE     byAlarmInfoType;
    BYTE     byRetAlarmTypeV40;
    BYTE     byRetDevInfoVersion;
    BYTE     byRetVQDArmType;
    BYTE     byRes[11];
}NET_DVR_SETUPALARM_PARAM, *LPNET_DVR_SETUPALARM_PARAM;
```

Members

dwSize

结构体大小

byLevel

布防等级，LCD 屏不支持，保留，设为 0

byAlarmInfoType

智能交通报警信息上传类型，LCD 屏不支持，保留，设为 0

byRetAlarmTypeV40

LCD 屏不支持，保留，设为 0

byRetDevInfoVersion

LCD 屏不支持，保留，设为 0

byRetVQDAlarmType

VQD 报警上传类型类型，LCD 屏不支持，保留，设为 0

byRes

保留，置为 0

See Also

[NET_DVR_SetupAlarmChan_V41](#)

7.81 NET_DVR_SINGLE_RS232:RS232 串口参数

```
struct{
    DWORD    dwBaudRate;
    BYTE     byDataBit;
    BYTE     byStopBit;
    BYTE     byParity;
    BYTE     byFlowcontrol;
    DWORD    dwWorkMode;
}NET_DVR_SINGLE_RS232, *LPNET_DVR_SINGLE_RS232;
```

Members

dwBaudRate

波特率(bps), 0- 50, 1- 75, 2- 110, 3- 150, 4- 300, 5- 600, 6- 1200, 7- 2400, 8- 4800, 9- 9600, 10- 19200, 11- 38400, 12- 57600, 13- 76800, 14- 115.2k

byDataBit

数据有几位: 0- 5 位, 1- 6 位, 2- 7 位, 3- 8 位

byStopBit

停止位: 0- 1 位, 1- 2 位

byParity

是否校验: 0- 无校验, 1- 奇校验, 2- 偶校验

byFlowcontrol

是否流控: 0- 无, 1- 软流控, 2- 硬流控

dwWorkMode

工作模式, 0- 窄带传输(232 串口用于 PPP 拨号), 1- 控制台(232 串口用于参数控制), 2- 透明通道, 3- ptz 模式 (审讯温湿度传感器), 4- 报警盒模式

See Also

[NET_DVR_RS232CFG_V30](#)

7.82 NET_DVR_SIREN_PARAM:警号参数结构体

```
struct{
    DWORD    dwSize;
    BYTE      byName[NAME_LEN];
    WORD      wDelay;
    BYTE      byRes1[2];
    DWORD     dwOverallEventJointSirenOn;
    DWORD     dwSubsystemEventJointSirenOn[MAX_ALARMHOST_SUBSYSTEM];
    BYTE      byRes2[448];
}NET_DVR_SIREN_PARAM, *LPNET_DVR_SIREN_PARAM;
```

Members

dwSize

结构体大小

byName

警号名称

wDelay

输出延迟，单位：秒。取值范围：0~3599s，0 表示一直有输出

byRes1

保留

dwOverallEventJointSirenOn

全局事件联动警号输出开启，事件类型按位表示，bit0-主机防拆，bit1-全局键盘紧急报警，bit2-交流电断电，bit3-电池电压低，bit4-电话线掉线，bit5-有线网络异常，bit6-无线网络异常，bit7-键盘 485 断线

dwSubsystemEventJointSirenOn

子系统事件联动警号输出开启。数组下标表示子系统号，每一位数组按位表示事件类型，bit0-紧急报警，bit1-布防，bit2-撤防。例如：dwSubsystemEventJointSirenOn[i]&0x01==1 表示设置子系统(i+1)紧急报警联动该警号输出

byRes2

保留

Remarks

- 子系统事件“布防”联动警号输出功能中，警号固定响 2 秒，此时 wDelay 参数无效。
- 紧急报警事件：子系统键盘上的紧急报警按钮被按下。

See Also

[NET_DVR_GetDVRConfig](#)

7.83 NET_DVR_STREAM_INFO:流信息

```
struct{
    DWORD    dwSize;
    BYTE      byID[STREAM_ID_LEN];
    DWORD     dwChannel;
    BYTE      byRes[32];
}NET_DVR_STREAM_INFO,*LPNET_DVR_STREAM_INFO;
```

Members*dwSize*

结构体大小

byID

流 ID，为字母、数字和"_"的组合。智能交通摄像机不支持流 ID 标识功能，设为 0

dwChannel

设备通道号

byRes

保留，置为 0

See Also[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDeviceConfig](#)**7.84 NET_DVR_STREAM_MODE_TYPE:取流方式**

```

struct{
    BYTE                                byGetStreamType;
    BYTE                                byRes[3];
    NET\_DVR\_STREAM\_TYPE\_UNION          uGetStream;
}NET_DVR_STREAM_MODE_TYPE,*LPNET_DVR_STREAM_MODE_TYPE;

```

Members*byGetStreamType*

取流方式：0- 直接从设备取流；1- 从流媒体取流；2- 通过 IPServer 获得 ip 地址后取流；3- 通过 IPServer 找到设备，再通过流媒体去设备的流；4- 通过流媒体由 URL 去取流；6- 通过结构体参数组装 URL 去取流（包含用户名和密码）

byRes

保留，置为 0

uGetStream

不同取流方式

See Also[NET_DVR_STREAM_SRC_INFO](#)**7.85 NET_DVR_STREAM_TYPE_UNION:取流方式联合体**

```

union{
    NET\_DVR\_DIRECT\_CONNECT\_CHAN\_INFO    struChanInfo;
    NET\_DVR\_PU\_STREAM\_URL                struStreamUrl;
    NET\_DVR\_PU\_STREAM\_URL\_CFG            struStreamUrlCfg;
}NET_DVR_STREAM_TYPE_UNION,*LPNET_DVR_STREAM_TYPE_UNION;

```

Members*struChanInfo*

IP 通道信息

struStreamUrl

URL 取流配置信息

struStreamUrlCfg

通过结构体参数构成 URL 去取流

See Also

[NET_DVR_STREAM_MODE_TYPE](#)

7.86 NET_DVR_STREAM_SRC_CFG:流的来源配置

```
struct{
    DWORD                                dwSize;
    NET\_DVR\_STREAM\_INFO                  struStreamID;
    NET\_DVR\_STREAM\_SRC\_INFO              struStreamSrcInfo;
}NET_DVR_STREAM_SRC_CFG,*LPNET_DVR_STREAM_SRC_CFG;
```

Members

dwSize

结构体大小

struStreamInfo

流信息

struStreamSrcInfo

流的来源信息

See Also

[NET_DVR_GetDeviceConfig](#)

7.87 NET_DVR_STREAM_SRC_INFO:流的来源信息

```
struct{
    DWORD                                dwSize;
    NET\_DVR\_STREAM\_MODE\_TYPE              struStreamSrcInfo;
}NET_DVR_STREAM_SRC_INFO,*LPNET_DVR_STREAM_SRC_INFO;
```

Members

dwSize

结构体大小

struStreamSrcInfo

流的来源信息

See Also

[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDeviceConfig](#)

7.88 NET_DVR_STRUCTHEAD: 头信息结构体

```
struct{
    WORD        wLength;
    BYTE        byVersion;
    BYTE        byRes;
}NET_DVR_STRUCTHEAD,*LPNET_DVR_STRUCTHEAD;
```

Members

wLength

结构长度， sizeof(NET_DVR_INVALID_DISK_PARAM) 、 sizeof(NET_DVR_MOUNT_DISK_PARAM) 、 sizeof(NET_DVR_FTPCFG_V40)或者 sizeof(NET_DVR_WD1_CFG)等等

byVersion

高低 4 位分别代表高低版本，后续根据版本和长度进行扩展，不同的版本的长度进行限制

byRes

保留

See Also

[NET_DVR_FTPCFG_V40](#)

7.89 NET_DVR_SWITCH_DAY_TIME: 开关机计划参数结构体

```
struct{
    BYTE                byEnable;
    BYTE                bySwitchType;
    BYTE                byRes1[2];
    NET\_DVR\_DAYTIME    struTimePoint;
    BYTE                byRes2[8];
}NET_DVR_SWITCH_DAY_TIME,*LPNET_DVR_SWITCH_DAY_TIME;
```

Members

byEnable

是否启用该时间点：0- 不启用，1- 启用

bySwitchType

开关机类型：0- 关机，1- 开机

byRes1

保留，置为 0

struTimePoint

开关机计划时间点

byRes2

保留，置为 0

See Also

[NET_DVR_TIME_SWITCH_CFG](#)

7.90 NET_DVR_SWITCH_TIME_POINT: 按时间点开关机参数结构体

```
struct{
    BYTE                byEnable;
    BYTE                bySwitchType;
    BYTE                byRes1[2];
    NET\_DVR\_SYSTEM\_TIME struTimePoint;
    BYTE                byRes2[16];
}NET_DVR_SWITCH_TIME_POINT,*LPNET_DVR_SWITCH_TIME_POINT;
```

Members

byEnable

是否启用该时间点：0- 不启用，1- 启用

bySwitchType

开关机类型：0- 关机，1- 开机

byRes1

保留，置为 0

struTimePoint

开关机时间点

byRes2

保留，置为 0

See Also

[NET_DVR_TIME_SWITCH_CFG](#)

7.91 NET_DVR_SYSTEM_TIME: 时间信息结构体

```
struct{
    WORD    wYear;
    WORD    wMonth;
    WORD    wDay;
    WORD    wHour;
    WORD    wMinute;
    WORD    wSecond;
    WORD    wMilliSec;
    BYTE    byRes[2];
}NET_DVR_SYSTEM_TIME,*LPNET_DVR_SYSTEM_TIME;
```

Members

wYear

年

wMonth

月

wDay

日

wHour

时
 wMinute
 分
 wSecond
 秒
 wMilliSec
 毫秒
 byRes
 保留

See Also

[NET_DVR_LOG_V30](#)、[NET_DVR_PLAY_BACK_BY_TIME](#)

7.92 NET_DVR_TIME:时间参数

```

struct{
    DWORD    dwYear;
    DWORD    dwMonth;
    DWORD    dwDay;
    DWORD    dwHour;
    DWORD    dwMinute;
    DWORD    dwSecond;
}NET_DVR_TIME, *LPNET_DVR_TIME;
  
```

Members

dwYear
 年
 dwMonth
 月
 dwDay
 日
 dwHour
 时
 dwMinute
 分
 dwSecond
 秒

See Also

[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)、[NET_DVR_FindDVRLog_V30](#)

7.93 NET_DVR_TIME_EX:时间参数

```

struct{
    WORD    wYear;
  
```

```

    BYTE    byMonth;
    BYTE    byDay;
    BYTE    byHour;
    BYTE    byMinute;
    BYTE    bySecond;
    BYTE    byRes;
}NET_DVR_TIME_EX, *LPNET_DVR_TIME_EX;

```

Members

wYear

年

byMonth

月

byDay

日

byHour

时

byMinute

分

bySecond

秒

byRes

保留

See Also

[NET_DVR_VIS_REGISTER_INFO](#)

7.94 NET_DVR_TIMEPOINT:时间点参数

```

struct{
    DWORD    dwMonth;
    DWORD    dwWeekNo;
    DWORD    dwWeekDate;
    DWORD    dwHour;
    DWORD    dwMin;
}NET_DVR_TIMEPOINT, *LPNET_DVR_TIMEPOINT;

```

Members

dwMonth

月：[0,11]取值分别表示第 1 个月到第 12 个月

dwWeekNo

周：0- 第 1 周，1- 第 2 周，2- 第 3 周，3- 第 4 周，4- 最后一周

dwWeekDate

星期：0- 星期日，1- 星期一，2- 星期二，3- 星期三，4- 星期四，5- 星期五，6- 星期六

dwHour

小时，开始时间：0~23，结束时间：1~23

dwMin

分：0~59

See Also

[NET_DVR_ZONEANDDST](#)

7.95 NET_DVR_TIME_SWITCH_CFG:时间点参数

```
struct
{
    DWORD                dwSize;
    NET\_DVR\_SWITCH\_DAY\_TIME struSwitchDayTime[MAX_DAYS][MAX_DAY_TIME_POINT]; //周开关机
    计划
    NET\_DVR\_SWITCH\_TIME\_POINT struSwitchTimePoint[MAX_TIME_POINT]; //按时间点开关机参数
    BYTE byRes[64];
}NET_DVR_TIME_SWITCH_CFG,*LPNET_DVR_TIME_SWITCH_CFG;
```

Members

dwSize

结构体大小

struSwitchDayTime

周开关机计划

struSwitchTimePoint

按时间点开关机参数

See Also

[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDeviceConfig](#)

7.96 NET_DVR_UNLOCK_RECORD_INFO:可视对讲开锁记录信息

```
struct{
    BYTE        byUnlockType;
    BYTE        byRes1[3];
    BYTE        byControlSrc[NAME_LEN];
    DWORD       dwPicDataLen;
    BYTE        *pImage;
    BYTE        byRes2[212];
}NET_DVR_UNLOCK_RECORD_INFO,*LPNET_DVR_UNLOCK_RECORD_INFO;
```

Members

byUnlockType

开锁方式，具体定义如下：

```
enum _UNLOCK_TYPE_ENUM_{
    ENUM_UNLOCK_PASSWORD        = 1,
    ENUM_UNLOCK_HIJACKING        = 2,
    ENUM_UNLOCK_CARD              = 3,
    ENUM_UNLOCK_HOUSEHOLDER      = 4,
```



```

    ENUM_UNLOCK_CENTER_PLATFORM = 5
}UNLOCK_TYPE_ENUM
ENUM_UNLOCK_PASSWORD
密码开锁
ENUM_UNLOCK_HIJACKING
劫持开锁
ENUM_UNLOCK_CARD
刷卡开锁
ENUM_UNLOCK_HOUSEHOLDER
户主开锁
ENUM_UNLOCK_CENTER_PLATFORM
中心平台开锁

```

byRes1

保留，置为 0

byControlSrc

操作发起源信息，刷卡开锁时为卡号，其余情况下为设备编号

dwPicDataLen

图片数据长度

plmage

缓冲区指针，保存图片数据，默认 jpeg 格式

byRes2

保留，置为 0

See Also

[NET_DVR_VIDEO_INTERCOM_EVENT_INFO_UINON](#)

7.97 **NET_DVR_USER_INFO_V40:** 用户信息结构体

struct{

```

    BYTE          sUserName[NAME_LEN];
    BYTE          sPassword[PASSWD_LEN];
    BYTE          byLocalRight[MAX_RIGHT];
    BYTE          byRemoteRight[MAX_RIGHT];
    DWORD         dwNetPreviewRight[MAX_CHANNUM_V40];
    DWORD         dwLocalRecordRight[MAX_CHANNUM_V40];
    DWORD         dwNetRecordRight[MAX_CHANNUM_V40];
    DWORD         dwLocalPlaybackRight[MAX_CHANNUM_V40];
    DWORD         dwNetPlaybackRight[MAX_CHANNUM_V40];
    DWORD         dwLocalPTZRight[MAX_CHANNUM_V40];
    DWORD         dwNetPTZRight[MAX_CHANNUM_V40];
    DWORD         dwLocalBackupRight[MAX_CHANNUM_V40];
    NET\_DVR\_IPADDR byChannel;
    BYTE          byMACAddr[MACADDR_LEN];
    BYTE          byPriority;
    BYTE          byAlarmOnRight;

```

```

    BYTE                byAlarmOffRight;
    BYTE                byBypassRight;
    BYTE                byRes[118];
}NET_DVR_USER_INFO_V40,*LPNET_DVR_USER_INFO_V40;

```

Members

sUserName

用户名，只能用 16 字节

sPassword

密码

byLocalRight

本地权限，数组取值：0- 无权限，1- 有权限，每位数组表示一种权限：

数组 0: 本地控制云台

数组 1: 本地手动录象

数组 2: 本地回放

数组 3: 本地设置参数

数组 4: 本地查看状态、日志

数组 5: 本地高级操作(升级，格式化，重启，关机)

数组 6: 本地查看参数

数组 7: 本地管理模拟和 IP Camera

数组 8: 本地备份

数组 9: 本地关机/重启

byRemoteRight

远程权限，数组取值：0- 无权限，1- 有权限，每位数组表示一种权限：

数组 0: 远程控制云台

数组 1: 远程手动录象

数组 2: 远程回放

数组 3: 远程设置参数

数组 4: 远程查看状态、日志

数组 5: 远程高级操作(升级，格式化，重启，关机)

数组 6: 远程发起语音对讲

数组 7: 远程预览

数组 8: 远程请求报警上传、报警输出

数组 9: 远程控制，本地输出

数组 10: 远程控制串口

数组 11: 远程查看参数

数组 12: 远程管理模拟和 IP Camera

数组 13: 远程关机/重启

dwNetPreviewRight

远程可以预览的通道，从前往后顺序排列，遇到 0xffffffff 后续均为无效

dwLocalRecordRight

本地可以录像的通道，从前往后顺序排列，遇到 0xffffffff 后续均为无效

dwNetRecordRight

远程可以录像的通道，从前往后顺序排列，遇到 0xffffffff 后续均为无效

dwLocalPlaybackRight

本地可以回放的通道，从前往后顺序排列，遇到 0xffffffff 后续均为无效

dwNetPlaybackRight

远程可以回放的通道，从前往后顺序排列，遇到 0xffffffff 后续均为无效

dwLocalPTZRight

本地可以 PTZ 的通道，从前往后顺序排列，遇到 0xffffffff 后续均为无效

dwNetPTZRight

远程可以 PTZ 的通道，从前往后顺序排列，遇到 0xffffffff 后续均为无效

dwLocalBackupRight

本地备份权限通道，从前往后顺序排列，遇到 0xffffffff 后续均为无效

struUserIP

用户 IP 地址绑定(为 0 时表示允许任何地址)

byMACAddr

物理地址绑定(为 00:00:00:00:00:00 时表示允许任何地址)

byPriority

优先级: 0xff-无, 0-低, 1-中, 2-高

无.....表示不支持优先级的设置

低.....默认权限: 包括本地和远程回放、本地和远程查看日志和状态、本地和远程关机/重启

中.....包括本地和远程控制云台、本地和远程手动录像、本地和远程回放、语音对讲和远程预览、本地备份、本地/远程关机/重启

高.....管理员(支持所有权限)

byAlarmOnRight

报警输入口布防权限: 0-无权限, 1-有权限

byAlarmOffRight

报警输入口撤防权限: 0-无权限, 1-有权限

byBypassRight

报警输入口旁路权限: 0-无权限, 1-有权限

byRes

保留, 置为 0

See Also

[NET_DVR_USER_V40](#)

7.98 NET_DVR_USER_V40: 用户参数配置(扩展)结构体

```
struct{
    DWORD                dwSize;
    DWORD                dwMaxUserNum;
    NET\_DVR\_USER\_INFO\_V40 struUser[MAX_USERNUM_V30];
}NET_DVR_USER_V40,*LPNET_DVR_USER_V40;
```

Members

dwSize

结构体大小

dwMaxUserNum

设备支持的最大用户个数（只读）

struUser

用户信息，struUser[0]~struUser[dwMaxUserNum-1]有效

See Also

[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDeviceConfig](#)

7.99 NET_VCA_BEHAVIOR_ABILITY:行为分析能力输出参数结构体

```
struct{
    DWORD    dwSize;
    DWORD    dwAbilityType;
    BYTE     byMaxRuleNum;
    BYTE     byMaxTargetNum;
    BYTE     bySupport;
    BYTE     byRes[5];
    DWORD    dwAbilityTypeEx;
}NET_VCA_BEHAVIOR_ABILITY,*LPNET_VCA_BEHAVIOR_ABILITY;
```

Members

dwSize

结构体大小

dwAbilityType

支持的行为分析能力类型，按位表示，具体定义如下：

```
enum _VCA_ABILITY_TYPE_{
    TRAVERSE_PLANE_ABILITY    = 0x01,        //穿越警戒面
    ENTER_AREA_ABILITY        = 0x02,        //进入区域
    EXIT_AREA_ABILITY         = 0x04,        //离开区域
    INTRUSION_ABILITY         = 0x08,        //入侵
    LOITER_ABILITY            = 0x10,        //徘徊
    LEFT_TAKE_ABILITY         = 0x20,        //丢包捡包
    PARKING_ABILITY           = 0x40,        //停车
    RUN_ABILITY               = 0x80,        //奔跑
    HIGH_DENSITY_ABILITY      = 0x100,       //区域内人员密度
    LF_TRACK_ABILITY          = 0x200,       //球机跟踪
    VIOLENT_MOTION_ABILITY    = 0x400,       //剧烈运动检测
    REACH_HIGHT_ABILITY       = 0x800,       //攀高检测
    GET_UP_ABILITY            = 0x1000,      //起身检测
    LEFT_ABILITY              = 0x2000,      //丢包检测
    TAKE_ABILITY              = 0x4000,      //捡包检测
    LEAVE_POSITION            = 0x8000,      //离岗
    TRAIL_ABILITY             = 0x10000,     //尾随
    KEY_PERSON_GET_UP_ABILITY = 0x20000,     //重点人员起身检测
    STANDUP_ABILITY          = 0x40000,     //起立
    FALL_DOWN_ABILITY         = 0x80000,     //倒地
    AUDIO_ABNORMAL_ABILITY    = 0x100000,    //声强突变
```

```

ADV_REACH_HEIGHT_ABILITY    = 0x200000,           //折线攀高
TOILET_TARRY_ABILITY        = 0x400000,           //如厕超时
YARD_TARRY_ABILITY          = 0x800000,           //放风场滞留
ADV_TRAVERSE_PLANE_ABILITY  = 0x1000000,          //折线警戒面
HUMAN_ENTER_ABILITY         = 0x10000000,         //人靠近 ATM，只在 ATM_PANEL 模式下支持
OVER_TIME_ABILITY           = 0x20000000,         //操作超时，只在 ATM_PANEL 模式下支持
STICK_UP_ABILITY            = 0x40000000,         //贴纸条，支持区域规则
INSTALL_SCANNER_ABILITY     = 0x80000000          //安装读卡器,支持区域规则
} VCA_ABILITY_TYPE

```

byMaxRuleNum

最大规则数

byMaxTargetNum

最大目标数

bySupport

支持的功能类型，按位表示。bySupport & 0x01 支持标定功能 byBFullChanNum

byRes

保留，置为 0

dwAbilityTypeEx

支持的行为分析能力类型，按位表示，具体定义如下：

```

enum _VCA_ABILITY_TYPE_EX{
    PEOPLENUM_CHANGE_ABILITY    = 0x00000002,       //人数变化检测
    SPACING_CHANGE_ABILITY      = 0x00000004,       //间距变化检测
    EVENT_COMBINED_ABILITY      = 0x00000008,       //组合规则事件
    EVENT_SIT_QUIETLY           = 0x00000010        //静坐事件
} VCA_ABILITY_TYPE

```

See Also

[NET_DVR_GetDeviceAbility](#)

7.100 NET_VCA_DEV_ABILITY:智能设备能力集结构体

```

struct{
    DWORD    dwSize;
    BYTE     byVCACHanNum;
    BYTE     byPlateChanNum;
    BYTE     byBBaseChanNum;
    BYTE     byBAdvanceChanNum;
    BYTE     byBFullChanNum;
    BYTE     byATMChanNum;
    BYTE     byPDCCChanNum;
    BYTE     byITSChanNum;
    BYTE     byBPrisonChanNum;
    BYTE     byFSnapChanNum;

```

```

BYTE    byFSnapRecogChanNum;
BYTE    byFRetrievalChanNum;
BYTE    bySupport;
BYTE    byFRecogChanNum;
BYTE    byBPPerimeterChanNum;
BYTE    byTPSChanNum;
BYTE    byTFSCChanNum;
BYTE    byFSnapBFullChanNum;
BYTE    byHeatMapChanNum;
BYTE    bySmartVehicleNum;
BYTE    bySmartHVTNum;
BYTE    bySmartNum;
BYTE    byVehicleNum;
BYTE    bySmartRoadDetectionNum;
BYTE    bySmartFaceDetectionNum;
BYTE    bySmartHeatMapNum;
BYTE    byRes[14];

```

```
}NET_VCA_DEV_ABILITY,*LPNET_VCA_DEV_ABILITY;
```

Members

dwSize

结构体大小

byVCAChanNum

智能通道个数

byPlateChanNum

车牌通道个数

byBBaseChanNum

行为基本版个数

byBAdvanceChanNum

行为高级版个数

byBFullChanNum

行为完整版个数

byATMChanNum;

智能 ATM 通道个数

byPDCCChanNum

人数统计通道个数

byITSChanNum

交通事件通道个数

byBPrisonChanNum

行为监狱版通道个数

byFSnapChanNum

人脸抓拍通道个数

byFSnapRecogChanNum

人脸抓拍和识别通道个数

byFRetrievalChanNum

人脸后检索个数

bySupport

能力，位与结果为 0 表示不支持，1 表示支持

bySupport&0x1，表示是否支持智能跟踪

bySupport&0x2，表示是否支持 128 路取流扩展

byFRecogChanNum

人脸识别通道个数

byBPPerimeterChanNum

行为监狱版(周界)通道个数

byTPSChanNum

交通诱导通道个数

byTFSCChanNum

道路违章取证（违章事件）通道个数

byFSnapBFullChanNum

人脸抓拍和行为分析通道个数

byHeatMapChanNum

热度图通道个数

bySmartVehicleNum

SMART 事件+车辆检测通道个数

bySmartHVTNum

SMART 事件+混行检测通道个数

bySmartNum

SMART 事件个数

byVehicleNum

车辆检测通道个数

bySmartRoadDetectionNum

SMART 事件+道路监控通道个数

bySmartFaceDetectionNum

SMART 事件+人脸侦测通道个数

bySmartHeatMapNum

SMART 事件+热度图通道个数

byRes[14]

保留，置为 0

See Also

[NET_DVR_GetDeviceAbility](#)

7.101 NET_DVR_VALID_PERIOD_CFG: 有效期参数结构体

```
struct{
    BYTE                byEnable;
    BYTE                byRes1[3];
    NET\_DVR\_TIME\_EX    struBeginTime;
    NET\_DVR\_TIME\_EX    struEndTime;
```

```

    BYTE                byRes2[32];
}NET_DVR_VALID_PERIOD_CFG,*LPNET_DVR_VALID_PERIOD_CFG;

```

Members

byEnable

是否启用该有效期：0- 不启用，1- 启用

byRes1

保留，置为 0

struBeginTime

有效期起始时间

struEndTime

有效期结束时间

byRes2

保留，置为 0

See Also

[NET_DVR_CARD_CFG](#)

7.102 NET_DVR_VIDEOINTERCOM_STREAM: 可视对讲流通道参数结构体

```

struct{
    DWORD    dwSize;
    BYTE     byVisDevID[MAX_NAMELEN];
    BYTE     byDeviceName[NAME_LEN];
    BYTE     bySourceType;
    BYTE     byRes[255];
}NET_DVR_VIDEOINTERCOM_STREAM,*LPNET_DVR_VIDEOINTERCOM_STREAM;

```

Members

dwSize

结构体大小

byVisDevID

可视对讲设备编号，即长号，设备类型为 IPC 时全部为 0

byDeviceName

设备名称

bySourceType

视频源类型：0- 无意义，1- IPC，2- DVR/DVS/NVR，3- 门口机，4- 围墙机，5- 二次确认机

byRes

保留，置为 0

Remarks

设备是否支持可视对讲流通道参数配置或者支持的参数能力，可以通过设备能力集进行判断，对应 IP 可视对讲主机能力集（接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：[IP_VIEW_DEV_ABILITY](#)），节点：[<VideoIntercomStream>](#)。

See Also

[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDVRConfig](#)

7.103 NET_DVR_VIDEO_CALL_COND:可视对讲信令处理条件参数

```
struct{
    DWORD    dwSize;
    BYTE      byRes[128];
}NET_DVR_VIDEO_CALL_COND, *LPNET_DVR_VIDEO_CALL_COND;
```

Members

dwSize
结构体大小

byRes
保留，置为 0

See Also

[NET_DVR_StartRemoteConfig](#)

7.104 NET_DVR_VIDEO_CALL_PARAM:可视对讲信令处理参数

```
struct{
    DWORD    dwSize;
    DWORD    dwCmdType;
    BYTE      byRes[128];
}NET_DVR_VIDEO_CALL_PARAM, *LPNET_DVR_VIDEO_CALL_PARAM;
```

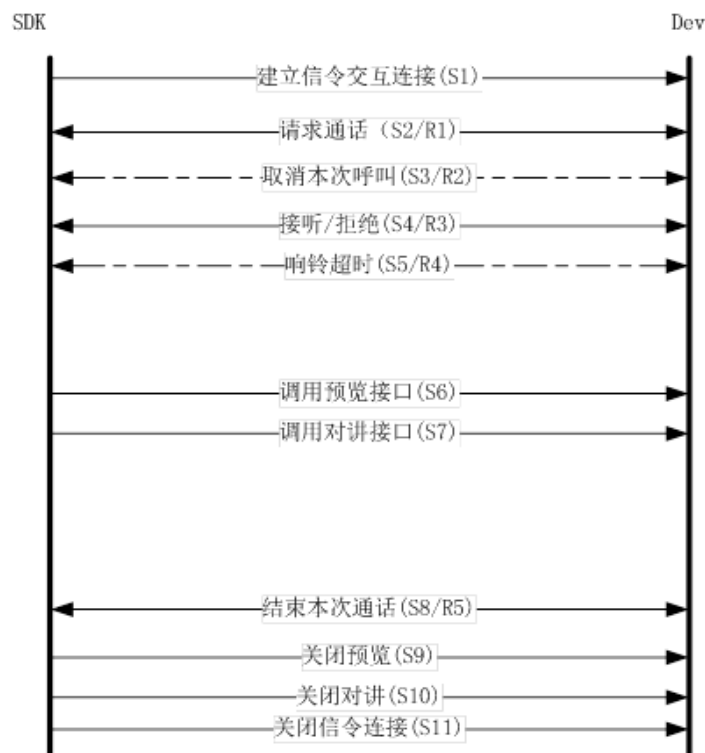
Members

dwSize
结构体大小

dwCmdType
信令类型：0- 请求呼叫，1- 取消本次呼叫，2- 接听本次呼叫，3- 拒绝本地来电呼叫，4- 被叫响铃超时，5- 结束本次通话，6- 设备正在通话中，7- 设备正在通话中

byRes
保留，置为 0

Remarks



- 可视通话能力，对应 IP 可视对讲主机能力集（接口：NET_DVR_GetDeviceAbility，能力集类型：IP_VIEW_DEV_ABILITY）中节点<VideoCall>。

- 该长连接配置接口结合报警、预览及对讲接口，可以完成可视通话的功能，具体流程机制如下所示：

客户端主动发起对讲：

1)当客户端主动发起连接请求后，客户端如果需要建立可视通话，则需要发送通话请求信令（S2），设备侧接收到该请求后，如果无人接听的情况下，30s 后设备端会返回响铃超时（R4），此时应用层直接关闭信令链接（S11）即可。

2)在客户端发送 S2 请求后，设备未返回 R3 及 R4 之前，客户端可以直接发送 S3，取消本次呼叫，然后直接调用 S7 即可。在客户端发送 S2 请求且设备做了应答后，则无法取消本次呼叫。

3)在客户端发送 S2 请求后，如果设备侧如果拒绝接听 R3，则客户端直接关闭连接即可。

4)在客户端发送 S2 请求后，如果设备接听，则客户端后续打开预览（不播放声音）、对讲即可进行视频通话（S6、S7）；如果中间任何一方想关闭连接，需要先要发送结束本次通话（S8/R5），后续关闭预览、对讲及信令连接即可。

设备侧主动发起对讲：

1)当客户端主动发起连接请求后，设备段如果需要建立可视通话，则需要发送通话请求信令（R1），客户端接收到该请求后，如果无人接听的情况下，30s 后会返回响铃超时（S5），此时应用层直接关闭信令链接（S11）即可。

2)在设备端发送 R1 请求后，客户端未返回 S4 及 S5 之前，设备端可以直接发送 R2，取消本次呼叫，然后直接调用 S7 即可。在设备端发送 R1 请求且客户端做了应答后，则无法取消本次呼叫。

3)在设备端发送 R1 请求后，如果客户如果拒绝接听 R3，则客户端直接关闭连接即可。

4)在设备侧发送 R1 请求后，如果客户端接听，则客户端后续打开预览（不播放声音）、对讲即可进行视频通话（S6、S7）；如果中间任何一方想关闭连接，需要先要发送结束本次通话（S8/R5），后续关闭预览、对讲及信令连接即可。

See Also

[NET_DVR_StartRemoteConfig](#)

7.105 NET_DVR_VIDEO_INTERCOM_ALARM:可视对讲报警信息

```
struct{
    DWORD                                dwSize;
    NET\_DVR\_TIME\_EX                      struTime;
    BYTE                                byDevNumber[MAX_DEV_NUMBER_LEN];
    BYTE                                byAlarmType;
    BYTE                                byRes1[3];
    NET\_DVR\_VIDEO\_INTERCOM\_ALARM\_INFO\_UNION uAlarmInfo;
    BYTE                                byRes2[256];
}NET_DVR_VIDEO_INTERCOM_ALARM, *LPNET_DVR_VIDEO_INTERCOM_ALARM;
```

Members

dwSize

结构体大小

struTime

报警时间

byDevNumber

设备编号

byAlarmType

报警类型: 1- 防区报警, 2- 防拆报警, 3- 劫持报警, 4- 多次密码开锁失败报警, 5- 门没开, 6- 门没关, 7- SOS(呼救报警)

byRes1

保留, 置为 0

uAlarmInfo

报警信息, *byAlarmType* 为 1 时有效

byRes2

保留, 置为 0

See Also

[NET_DVR_StartListen_V30](#)

7.106 NET_DVR_VIDEO_INTERCOM_ALARM_INFO_UNION:可视对讲报警信息联合体

```
union{
    BYTE                                byLen[256];
    NET\_DVR\_ZONE\_ALARM\_INFO             struZoneAlarm;
}NET_DVR_VIDEO_INTERCOM_ALARM_INFO_UNION, *LPNET_DVR_VIDEO_INTERCOM_ALARM_INFO_UNION;
```

Members

byLen

联合体大小, 256 字节

struZoneAlarm

防区报警信息，报警类型为“1-防区报警”时有效

See Also

[NET_DVR_VIDEO_INTERCOM_ALARM](#)

7.107 **NET_DVR_VIDEO_INTERCOM_ALARM_CFG**: 可视对讲报警事件参数结构体

```
struct{
    DWORD    dwSize;
    BYTE     byDoorNotCloseAlarm;
    BYTE     byRes[603];
}NET_DVR_VIDEO_INTERCOM_ALARM_CFG, *LPNET_DVR_VIDEO_INTERCOM_ALARM_CFG;
```

Members

dwSize

结构体大小

byDoorNotCloseAlarm

是否上传门未关报警：0- 不上传，1- 上传

byRes

保留，置为 0

Remarks

设备是否支持可视对讲报警事件参数配置或者支持的参数能力，可以通过设备能力集进行判断，对应 IP 可视对讲主机能力集（接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：[IP_VIEW_DEV_ABILITY](#)），节点：[<UploadAlarmCfg>](#)。

See Also

[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDVRConfig](#)

7.108 **NET_DVR_VIDEO_INTERCOM_DEVICEID_CFG**: 可视对讲设备编号配置结构体

```
struct{
    DWORD                                dwSize;
    BYTE                                byUnitType;
    BYTE                                byIsAutoReg;
    BYTE                                byRes1[2];
    NET\_DVR\_VIDEO\_INTERCOM\_UNIT\_DEVICEID\_UNION uVideoIntercomUnit;
    BYTE                                byRes2[128];
}NET_DVR_VIDEO_INTERCOM_DEVICEID_CFG,*LPNET_DVR_VIDEO_INTERCOM_DEVICEID_CFG;
```

Members

dwSize

结构体大小

byUnitType

设备类型：1- 门口机，2- 管理机，4- 围墙机，5- 别墅门口机，6- 二次确认机

byIsAutoReg

是否自动注册：0- 否，1- 是

byRes1

保留，置为 0

uVideoIntercomUnit

设备编号参数，不同的设备类型对应不同的结构

byRes2

保留，置为 0

Remarks

设备编号参数能力，对应 IP 可视对讲主机能力集（接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：[IP_VIEW_DEV_ABILITY](#)）中节点<DeviceID>。二次确认机不配置编号。

See Also

[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDVRConfig](#)

7.109 NET_DVR_VIDEO_INTERCOM_EVENT: 可视对讲事件记录信息结构体

```
struct{
    DWORD                                dwSize;
    NET\_DVR\_TIME\_EX                     struTime;
    BYTE                                byDevNumber[MAX_DEV_NUMBER_LEN];
    BYTE                                byEventType;
    BYTE                                byRes1[3];
    NET\_DVR\_VIDEO\_INTERCOM\_EVENT\_INFO\_UNION uEventInfo;
    BYTE                                byRes2[256];
}NET_DVR_VIDEO_INTERCOM_EVENT, *LPNET_DVR_VIDEO_INTERCOM_EVENT;
```

Members

dwSize

结构体大小

struTime

事件记录时间

byDevNumber

设备编号

byEventType

事件信息类型：1- 开锁记录，2- 公告信息阅读回执

byRes1

保留，置为 0

uEventInfo

事件信息，不同的类型对应不同的结构

byRes2

保留，置为 0

See Also

[NET_DVR_StartListen_V30](#)

7.110 NET_DVR_VIDEO_INTERCOM_EVENT_INFO_UINON: 可视对讲事件记录信息联合体

```
union{
    BYTE                                     byLen[256];
    NET\_DVR\_UNLOCK\_RECORD\_INFO             struUnlockRecord;
    NET\_DVR\_NOTICEDATA\_RECEIPT\_INFO         struNoticedataReceipt;
}NET_DVR_VIDEO_INTERCOM_EVENT_INFO_UINON, *LPNET_DVR_VIDEO_INTERCOM_EVENT_INFO_UINON;
```

Members

byLen

联合体大小，256 字节

struUnlockRecord

开锁记录

struNoticedataReceipt

公告信息阅读回执

See Also

[NET_DVR_VIDEO_INTERCOM_EVENT](#)

7.111 NET_DVR_VIDEO_INTERCOM_IOIN_CFG: 可视对讲 IO 输入参数配置结构体

```
struct{
    DWORD    dwSize;
    BYTE     byIOUseType;
    BYTE     byRes[63];
}NET_DVR_VIDEO_INTERCOM_IOIN_CFG, *LPNET_DVR_VIDEO_INTERCOM_IOIN_CFG;
```

Members

dwSize

结构体大小

byIOUseType

IO 输入用途：0- 禁用，1- 开门按钮，2- 门状态，0xff- 自定义

byRes

保留，置为 0

Remarks

IO 输入参数配置能力，对应 IP 可视对讲主机能力集（接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：

[IP_VIEW_DEV_ABILITY](#)) 中节点<IOIn>。

See Also

[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDVRConfig](#)

7.112 NET_DVR_VIDEO_INTERCOM_IOOUT_CFG:可视对讲 IO 输出参数配置结构体

```
struct{
    DWORD    dwSize;
    BYTE     byIOUseType;
    BYTE     byRes[63];
}NET_DVR_VIDEO_INTERCOM_IOOUT_CFG, *LPNET_DVR_VIDEO_INTERCOM_IOOUT_CFG;
```

Members

dwSize

结构体大小

byIOUseType

IO 输出用途：0- 禁用，1- 电锁，0xff- 自定义

byRes

保留，置为 0

Remarks

IO 输出参数配置能力，对应 IP 可视对讲主机能力集（接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：[IP_VIEW_DEV_ABILITY](#)）中节点<IOOut>。

See Also

[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDVRConfig](#)

7.113 NET_DVR_VIDEO_INTERCOM_UNIT_DEVICEID_UNION：可视对讲设备编号参数联合体

```
union{
    BYTE                                     byLen[128];
    NET\_DVR\_OUTDOOR\_UNIT\_DEVICEID          struOutdoorUnit;
    NET\_DVR\_MANAGE\_UNIT\_DEVICEID           struManageUnit;
    NET\_DVR\_OUTDOOR\_FENCE\_DEVICEID         struFenceUnit;
    NET\_DVR\_OUTDOOR\_UNIT\_DEVICEID          struVillaOutdoorUnit;
}NET_DVR_VIDEO_INTERCOM_UNIT_DEVICEID_UNION,*LPNET_DVR_VIDEO_INTERCOM_UNIT_DEVICEID_UNION;
```

Members

byLen

联合体大小，128 字节

struOutdoorUnit

门口机设备编号参数

struManageUnit

管理机设备编号参数

struFenceUnit

围墙机设备编号参数

struVillaOutdoorUnit

别墅门口机设备编号参数

Remarks

设备根据配置信息，生成设备编号字符串，规则如下：

- 1) 管理机的长号为*00000001XX；*代表期号，001 代表管理机，XX 代表管理机编号(从 01 开始，01、02、...，加 1 递增)。
- 2) 门口机或者别墅门口机的编号为*XXXXX000\$\$；*代表期号，XXX 代表楼号，XX 单元号，000 代表门口机，\$\$代表门口机序号，序号为 00 则代表主门口机，其他取值则代表副门口机。
- 3) 围墙机的长号为*00000002XX；*代表期号，002 代表围墙机，XX 代表围墙机编号(从 01 开始，01、02、...，加 1 递增)。

实际应用时同一管理机可能属于多期楼房，对于此种情况，管理机的期号赋值可以为任何一个存在期号，只要保证期号、管理机序号及其他组合信息生成的设备编号是唯一的即可。

See Also

[NET_DVR_VIDEO_INTERCOM_DEVICEID_CFG](#)

7.114 NET_DVR_VIDEO_INTERCOM_OPERATION_TIME_CFG：操作时间配置结构体

```
struct{
    DWORD                dwSize;
    BYTE                 byUnitType;
    BYTE                 byRes1[3];
    NET\_DVR\_VIDEO\_INTERCOM\_OPERATION\_TIME\_UNION uVideoIntercomUnit;
    BYTE                 byRes2[128];
}NET_DVR_VIDEO_INTERCOM_OPERATION_TIME_CFG, *LPNET_DVR_VIDEO_INTERCOM_OPERATION_TIME_CFG;
```

Members

dwSize

结构体大小

byUnitType

设备类型：1- 门口机，2- 管理机，4- 围墙机，5- 别墅门口机，6- 二次确认机

byRes1

保留，置为 0

uVideoIntercomUnit

操作时间配置联合体，不同的设备类型对应不同的结构

byRes2

保留，置为 0

Remarks

操作时间配置能力，对应 IP 可视对讲主机能力集（接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：[IP_VIEW_DEV_ABILITY](#)）中节点<OperationTime>。

See Also

[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDVRConfig](#)

7.115 NET_DVR_VIDEO_INTERCOM_OPERATION_TIME_UNION：可视对讲操作时间配置联合体

```
union{
    BYTE                                     byLen[128];
    NET\_DVR\_OUTDOOR\_UNIT\_OPERATION\_TIME\_CFG struOutdoorUnit;
    NET\_DVR\_MANAGE\_UNIT\_OPERATION\_TIME\_CFG struManageUnit;
}NET_DVR_VIDEO_INTERCOM_OPERATION_TIME_UNION,
*LPNET_DVR_VIDEO_INTERCOM_OPERATION_TIME_UNION;
```

Members

byLen

联合体大小，128 字节

struOutdoorUnit

门口机/围墙机/别墅门口机/二次确认机操作时间配置

struManageUnit

管理机操作时间配置

See Also

[NET_DVR_VIDEO_INTERCOM_OPERATION_TIME_CFG](#)

7.116 NET_DVR_VIDEO_INTERCOM_RELATEDDEV_CFG：可视对讲关联网络设备配置结构体

```
struct{
    DWORD                                     dwSize;
    BYTE                                     byUnitType;
    BYTE                                     byRes1[3];
    NET\_DVR\_VIDEO\_INTERCOM\_UNIT\_RELATEDDEV\_UNION uVideoIntercomUnit;
    BYTE                                     byRes2[128];
}NET_DVR_VIDEO_INTERCOM_RELATEDDEV_CFG, *LPNET_DVR_VIDEO_INTERCOM_RELATEDDEV_CFG;
```

Members

dwSize

结构体大小

byUnitType

设备类型：1- 门口机，2- 管理机，4- 围墙机，5- 别墅门口机，6- 二次确认机

byRes1

保留，置为 0

uVideoIntercomUnit

关联网络设备配置联合体，不同的设备类型对应不同的结构

byRes2

保留，置为 0

Remarks

关联网络设备配置能力，对应 IP 可视对讲主机能力集（接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：[IP_VIEW_DEV_ABILITY](#)）中节点<RelateDevice>。

See Also

[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDVRConfig](#)

7.117 NET_DVR_VIDEO_INTERCOM_UNIT_RELATEDDEV_UNION：可视对讲关联网络设备配置联合体

```
union{
    DWORD                                dwRes[256];
    NET\_DVR\_OUTDOOR\_UNIT\_RELATEDDEV      struOutdoorUnit;
    NET\_DVR\_MANAGE\_UNIT\_RELATEDDEV       struManageUnit;
    NET\_DVR\_OUTDOOR\_UNIT\_RELATEDDEV       struVillaUnit;
    NET\_DVR\_AGAIN\_RELATEDDEV             struAgainUnit;
}NET_DVR_VIDEO_INTERCOM_UNIT_RELATEDDEV_UNION,
*LPNET_DVR_VIDEO_INTERCOM_UNIT_RELATEDDEV_UNION;
```

Members

dwRes

联合体大小，256*4 共 1024 字节

struOutdoorUnit

门口机/围墙机关联网络设备参数配置

struManageUnit

管理机关联网络设备参数配置

struVillaUnit

别墅门口机关联网络设备参数配置

struAgainUnit

二次确认机关联网络设备参数配置

See Also

[NET_DVR_VIDEO_INTERCOM_RELATEDDEV_CFG](#)

7.118 NET_DVR_VILOST:信号丢失报警参数结构体

```
struct{
    BYTE                                byEnableHandleVILost;
```

```

NET\_DVR\_HANDLEEXCEPTION    strVILostHandleType;
NET\_DVR\_SCHEDTIME          struAlarmTime[MAX_DAYS][MAX_TIMESEGMENT];
}NET_DVR_VILOST, *LPNET_DVR_VILOST;

```

Members

byEnableHandleVILost

是否处理信号丢失报警：0-不处理，1-处理

strVILostHandleType

处理方式参数

struAlarmTime

布防时间参数

See Also

[NET_DVR_PICCFG](#)

7.119 NET_DVR_VIS_REGISTER_INFO: 门口机注册信息结构体

```

struct{
    DWORD          dwSize;
    DWORD          dwID;
    BYTE           szDevNumber[MAX_DEV_NUMBER_LEN];
    DWORD          byMACAddr[MACADDR_LEN];
    BYTE           byRes1[2];
    DWORD          sSerialNumber[SERIALNO_LEN];
    NET\_DVR\_IPADDR  struDevIP;
    NET\_DVR\_TIME\_EX struRegisterTime;
    BYTE           byRegisterType;
    BYTE           byRes[127];
}NET_DVR_VIS_REGISTER_INFO, *LPNET_DVR_VIS_REGISTER_INFO;

```

Members

dwSize

结构体大小

dwID

注册 ID

szDevNumber

设备编号

byMACAddr

Mac 地址

byRes1

保留，置为 0

sSerialNumber

设备序列号

struDevIP

设备 IP 地址

struRegisterTime

注册时间

byRegisterType

注册类型: 0- 保留, 1- 门口机, 2- 管理机, 4- 围墙机, 5- 别墅门口机, 6- 二次确认机, 7- 8700 客户端, 8- 4200 客户端

byRes

保留, 置为 0

See Also

[NET_DVR_GetNextRemoteConfig](#)

7.120 NET_DVR_VCS_USER_INFO: 用户信息结构体

```
struct{
    DWORD                dwSize;
    BYTE                sUserName[NAME_LEN];
    BYTE                sPassWord[PASSWD_LEN];
    NET\_DVR\_IPADDR      struUserIP;
    BYTE                byMacAddr[MACADDR_LEN];
    BYTE                byPriority;
    BYTE                byRes1;
    DWORD                dwRight;
    NET\_DVR\_ASSOCIATE\_INPUT\_PARAM struInputParam[MAX_INPUTNUMS];
    NET\_DVR\_ASSOCIATE\_OUTPUT\_PARAM struOutputParam[MAX_OUTPUTNUMS];
    BYTE                byRes2[128];
}NET_DVR_VCS_USER_INFO,*LPNET_DVR_VCS_USER_INFO;
```

Members

dwSize

结构体大小

sUserName

用户名, 最大 16 字节

sPassWord

密码

struUserIP

绑定用户 IP 地址(为 0 时表示允许任何地址)

byMacAddr

物理地址绑定(为 0 时表示允许任何 MAC 地址)

byPriority

优先级: 1-管理员, 2-操作员, 管理员具有任何权限, 操作员没有"管理用户"的权限

byRes1

保留, 置为 0

dwRight

远程操作权限, 按位表示: 0-不支持, 1-支持, 各 32 位代表的权限类型如下所示:

bit0-获取设置基本参数, bit1-显示屏管理, bit2-窗口管理, bit3-信号源管理,

bit4-布局管理, bit5-预案管理, bit6-底图管理, bit7-OSD 管理, bit8-回显权限,

bit9-报警布防, bit10-远程升级, bit11-恢复默认设置, bit12-导入导出配置,
bit13-日志查询, bit14-远程重启, bit15-屏幕区域管理

struInputParam

关联的输入设备

struOutputParam

关联的输出显示屏

byRes2

保留, 置为 0

Remarks

- 多屏控制器具有一个 ADMIN 管理员用户和 7 个普通用户: 管理员登陆时, 普通用户不能操作; 管理员没有登陆时, 几个普通用户可以同时操作。管理员具有所有操作权限 (包括屏幕和设备), 能够添加、修改、删除普通用户信息。
- 参数 struUserIP、byMacAddr、struInputParam、struOutputParam 在集中式多屏控制器 V2.1 中无效。
- 设置操作员用户参数时, 用户名 (sUserName) 为空表示删除相应用户。

See Also

[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)

7.121 NET_DVR_VIDEOEFFECT: 视频参数

```
struct{
    BYTE    byBrightnessLevel;
    BYTE    byContrastLevel;
    BYTE    bySharpnessLevel;
    BYTE    bySaturationLevel;
    BYTE    byHueLevel;
    BYTE    byEnableFunc;
    BYTE    byLightInhibitLevel;
    BYTE    byGrayLevel;
}NET_DVR_VIDEOEFFECT, *LPNET_DVR_VIDEOEFFECT;
```

Members

byBrightnessLevel

亮度, 取值范围: [0,100]

byContrastLevel

对比度, 取值范围: [0,100]

bySharpnessLevel

锐度, 取值范围: [0,100]

bySaturationLevel

饱和度, 取值范围: [0,100]

byHueLevel

色度, 取值范围: [0,100], 保留

byEnableFunc

使能, 按位表示。bit0-SMART IR(防过曝), bit1-低照度, bit2-强光抑制使能, 值: 0-否, 1-是, 例如 byEnableFunc&0x2==1 表示使能低照度功能; bit3-锐度类型, 值: 0-自动, 1-手动。

byLightInhibitLevel

强光抑制等级，取值范围：[1,3]

byGrayLevel

灰度值域:0-[0,255]， 1-[16,235]

7.122 NET_DVR_VOLUME_CFG:音量调节参数配置

```
struct{
    DWORD    dwSize;
    WORD     wVolume[MAX_AUDIOOUT_PRO_TYPE];
    BYTE     byPhantomPowerSupply;
    BYTE     byEnableAEC;
    BYTE     byRes1[2];
    BYTE     byEnableFBC[MAX_AUDIOOUT_PRO_TYPE];
    BYTE     byRes[20];
}NET_DVR_VOLUME_CFG,*LPNET_DVR_VOLUME_CFG;
```

Members

dwSize

结构体大小

wVolume

音量大小，数组 0 表示音频输出，数组 1 表示音频编码，具体索引代表含义以能力集为准

byPhantomPowerSupply

是否使用幻象电源供电(音频输入通道为 MIC 时有效): 0- 无意义，1- 不供电，2- 供电

byEnableAEC

是否启用全局的回声消除: 0- 不启用，1- 启用

byRes1

保留，置为 0

byEnableFBC

是否启用 FBC(啸叫抑制): 0- 不启用，1- 启用

byRes

保留，置为 0

See Also

[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDVRConfig](#)

7.123 NET_DVR_WDR:宽动态参数结构体

```
struct{
    BYTE     byWDREnabled;
    BYTE     byWDRLevel1;
    BYTE     byWDRLevel2;
    BYTE     byWDRContrastLevel;
    BYTE     byRes[16];
}NET_DVR_WDR, *LPNET_DVR_WDR;
```

Members

byWDREnabled

宽动态是否启用, 0-不启用, 1-启用, 2-自动

byWDRLevel1

0-F

byWDRLevel2

0-F

byWDRContrastLevel

0-100

byRes

保留

See Also[NET_DVR_WDR_CFG](#)**7.124 NET_DVR_WDR_CFG:宽动态参数配置结构体**

```
struct{
    DWORD          dwSize;
    NET\_DVR\_WDR    struWDR;
}NET_DVR_WDR_CFG, *LPNET_DVR_WDR_CFG;
```

Members

dwSize

结构体大小

struWDR

宽动态参数

Remarks

宽动态参数能力, 对应 IP 可视对讲主机能力集 (接口: [NET_DVR_GetDeviceAbility](#), 能力集类型: [IP_VIEW_DEV_ABILITY](#)) 能力集类型: DEVICE_ABILITY_INFO。

See Also[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDVRConfig](#)**7.125 NET_DVR_ZONE_ALARM_INFO:防区报警信息**

```
struct{
    BYTE          byZoneName[NAME_LEN];
    DWORD         dwZonendex;
    BYTE          byZoneType;
    BYTE          byRes[219];
}NET_DVR_ZONE_ALARM_INFO, *LPNET_DVR_ZONE_ALARM_INFO;
```

Members

byZoneName

防区名称

dwZonendex

防区号

byZoneType

防区类型，具体定义如下：

```
enum _ALARM_ZONE_TYPE_ENUM_{
    ENUM_ALARM_ZONE_TYPE_MANUAL          = 0,
    ENUM_ALARM_ZONE_TYPE_MAGNETIC        = 1,
    ENUM_ALARM_ZONE_TYPE_SMOKE           = 2,
    ENUM_ALARM_ZONE_TYPE_ACTIVE_INFRARED = 3,
    ENUM_ALARM_ZONE_TYPE_PASSIVE_INFRARED = 4,
    ENUM_ALARM_ZONE_TYPE_GAS             = 11
}ALARM_ZONE_TYPE_ENUM
```

ENUM_ALARM_ZONE_TYPE_MANUAL

紧急开关报警

ENUM_ALARM_ZONE_TYPE_MAGNETIC

门磁报警

ENUM_ALARM_ZONE_TYPE_SMOKE

烟感报警

ENUM_ALARM_ZONE_TYPE_ACTIVE_INFRARED

主动红外报警

ENUM_ALARM_ZONE_TYPE_PASSIVE_INFRARED

被动红外报警

ENUM_ALARM_ZONE_TYPE_GAS

煤气报警

byRes

保留，置为 0

See Also

[NET_DVR_VIDEO_INTERCOM_ALARM_INFO_UNION](#)

7.126 **NET_DVR_ZONEANDDST**:夏令时参数

```
struct{
    DWORD          dwSize;
    DWORD          dwZoneIndex;
    BYTE           byRes1[12];
    DWORD          dwEnableDST;
    BYTE           byDSTBias;
    BYTE           byRes2[3];
    NET\_DVR\_TIMEPOINT struBeginPoint;
    NET\_DVR\_TIMEPOINT struEndPoint;
}NET_DVR_ZONEANDDST,*LPNET_DVR_ZONEANDDST;
```

Members

dwSize

结构体大小

dwZoneIndex

该参数请赋值为 [NET_DVR_GetDVRConfig](#) 获取的值，设置时区请通过设置 [NET_DVR_NTTPARA](#) 中的 cTimeDifferenceH 或 cTimeDifferenceM 来实现

byRes1

保留，置为 0

dwEnableDST

是否启用夏时制：0- 不启用，1- 启用

byDSTBias

夏令时偏移值（以分钟计）：30min，60min，90min，120min

byRes2

保留，置为 0

struBeginPoint

夏时制开始时间

struEndPoint

夏时制停止时间

See Also

[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)

7.127 IP 可视对讲主机能力集能力 XML 描述

```
<?xml version="1.0" encoding="utf-8" ?>
<!-- req, IP 可视对讲主机设备能力集描述 -->
<IpViewDevAbility version="2.0">
  <!-- req,SIP 服务器注册能力 -->
  <SipServerLogin>

    <!--req,支持自动注册 -->
    <AutoLogin>
      <enable>true</enable>
    </AutoLogin>

    <!--req,注册状态-->
    <loginStatus opt="registered,unregistered" />
    <!--req,注册用户名长度-->
    <sipLoginNameLen min="1" max="32" />
    <!-- req,注册密码长度-->
    <sipLoginPasswordLen min="1" max="16" />
    <!-- req,设备显示名称长度-->
    <displayNameLen min="1" max="128" />
    <!-- req,本机号码长度-->
    <localNumber min="1" max="32" />
    <!-- req, 注册周期,单位: 分钟 -->
    <loginCycle min="1" max="99" />
  </SipServerLogin>
</IpViewDevAbility>
```

```

</SipServerLogin>

<!-- req,对讲分机基本能力 -->
<LocalAbility>
  <!-- req,本机默认铃音可选范围 -->
  <defaultRing min="1" max="6" />
  <!-- req,本机铃音音量范围 -->
  <ringVolume min="1" max="9" />
  <!-- req, 输入音量可选范围 -->
  <inputVolume min="0" max="6" />
  <!-- req, 输出音量可选范围 -->
  <outputVolume min="0" max="9" />
  <!-- req, 支持的音频编码级数 -->
  <audioEncPriNum>2</audioEncPriNum>
  <!-- req,预览延时时延范围, 单位: 秒 -->
  <delayPreview min="0" max="30" />

  <!-- req, 支持的音频编码类型,同级节点可有多个 -->
  <AudioEncEntry>
    <index>0</index>
    <name>OggVorbis</name>
    <packetLen opt="160,320" />
  </AudioEncEntry>

  <!-- req, 呼叫能力 -->
  <CallAbility>

    <!-- req,支持自动应答 -->
    <AutoResponse>
      <!-- req,自动应答时间范围, 单位: 秒 -->
      <autoResponse min="0" max="30" />
    </AutoResponse>

    <!-- req,支持的呼叫号码数 -->
    <callNumber>2</callNumber>
    <callNumberLen min="0" max="32" />
  </CallAbility>
</LocalAbility>

<!-- req,可视对讲, 门口机、管理机、别墅门口机、二次确认机 -->
<VideoIntercom>
  <monitorChannelNo min="1" max="100" />
  <!-- req,管理机的监控通道 -->
  <DeviceID>

```

```

<!-- req,设备编号 -->
    <enabled>true</enabled>
<!-- req,支持设备编号配置 -->
    <supportUnitType opt="" />
<!-- req,支持的对讲机类型 -->

<OutDoorDevice>
<!-- req,门口机 -->
    <period min="1" max="16" />
<!-- req,期号 -->
    <buildingNum min="1" max="16" />
<!-- req,楼幢号 -->
    <unitNum min="0" max="10" />
<!-- req,单元号 -->
    <floorNum min="1" max="16" />
<!-- req,层号 -->
    <devIndex min="0" max="10" />
<!-- req,门口机编号 -->
</OutDoorDevice>

<ManageUnitDevice>
<!-- req,管理机 -->
    <period min="1" max="16" />
<!-- req,期号 -->
    <devIndex min="0" max="10" />
<!-- req,管理机编号 -->
</ManageUnitDevice>

<OutDoorFenceDevice>
<!-- req,围墙机 -->
    <period min="1" max="16" />
<!-- req,期号 -->
    <devIndex min="0" max="10" />
<!-- req,围墙机编号 -->
</OutDoorFenceDevice>

<VillaOutDoorDevice>
<!-- req 别墅门口机 -->
    <period min="1" max="16" />
<!-- req,期号 -->
    <buildingNum min="1" max="16" />
<!-- req,楼幢号 -->
    <unitNum min="0" max="10" />
<!-- req,单元号 -->

```

```

    <floorNum min="1" max="16" />
    <!-- req,层号 -->
    <devIndex min="0" max="10" />
    <!-- req,门口机编号 -->
</VillaOutDoorDevice>

<AgainDevice>
    <!-- req,二次确认机 -->
</AgainDevice>
</DeviceID>

<!-- req,权限密码 -->
<PrivilegePwd>
    <pwdType opt="engineering,setupAlarm,householderUnlock,antiHijacking" />
    <!-- req,密码类型, engineering-工程密码,setupAlarm-布撤防密码,householderUnlock-户主开锁密码,antiHijacking-防劫持密码 -->
    <pwdLen min="6" max="16" />
    <!-- 密码长度 -->
</PrivilegePwd>

<!-- req,操作时间 -->
<OperationTime>
    <monitoringTime min="10" max="60" />
    <!-- req,最大监视时间, 单位: 秒 -->
    <ringTime min="15" max="60" />
    <!-- req,最大振铃时间, 单位: 秒 -->
    <messageTime min="30" max="60" />
    <!-- req,最大留言时间, 单位: 秒 -->
    <talkTime min="90" max="120" />
    <!-- req,最大通话时间, 单位: 秒 -->
    <callForwardingTime min="0" max="20" />
    <!-- 呼叫转移超时时间, 单位: 秒 -->
</OperationTime>

<RelateDevice>
    <outdoorUnitIP opt="ipv4,ipv6" />
    <!-- req,(主) 门口机 IP -->
    <manageUnitIP opt="ipv4,ipv6" />
    <!-- req,管理机 IP -->
    <sipServerIP opt="ipv4,ipv6" />
    <!-- req,SIP 服务器 IP -->
    <centerIP opt="ipv4,ipv6" />
    <!-- req,中心平台 IP -->
    <centerPort min="0" max="65535" />

```

```

<!-- req,中心平台端口 -->
<notSupportAgainUnitIP opt="true,false" />
<!-- req,是否不支持配置二次确认机 IP, true-不支持, false-支持 -->
<againUnitIP opt="ipv4,ipv6" />
<!-- req,二次确认机 IP -->
<notSupportOutdoorType opt="true,false" />
<!-- req,是否不支持配置主门口机类型, true-不支持, false-支持 -->
<outDoorType opt="unitOutdoor,villaOutDoor" />
<!-- req,主门口机类型, unitOutdoor-主单元门口机, villaOutDoor-主别墅门口机 -->
</RelateDevice>

<NoticeData>
  <noticeTime>true</noticeTime>
  <!-- req,公告时间 -->
  <noticeNumberLen min="0" max="32" />
  <!-- req,公告编号 -->
  <noticeThemeLen min="0" max="64" />
  <!-- req,公告主题 -->
  <noticeDetailLen min="0" max="1024" />
  <!-- req,公告详情 -->
  <noticeLevel opt="advertisement,propertyMgmt,alarm,notification" />
  <!-- req,公告等级, advertisement-广告类信息,propertyMgmt-物业信息,alarm-报警信息,notification-通知类信息-->
  <maxPicNum>6</maxPicNum>
  <!-- req,图片数量 -->
  <maxSinglePicSize>512</maxSinglePicSize>
  <!-- req,单张图片的最大大小, 单位: KB -->
</NoticeData>

<!-- req,远程开锁 -->
<ControlGateway>
  <gatewayIndex min="1" />
  <!-- req,门禁序号 -->
  <command opt="close,open" />
  <!-- req,控制命令 -->
  <controlSrc>true</controlSrc>
  <!-- req,操作发起源信息 -->
</ControlGateway>

<!-- 防区配置 -->
<Zone>
  <ZoneConfig>
    <enabled>true</enabled>
    <!-- req,支持防区参数配置 (报警输入参数配置) -->

```

```

<delayInParam>true</delayInParam>
<!-- req,true,表示客户端应该使用 NET_DVR_ALARMING_PARAM 中的 dwParam 来设置延时时间,动环
报警主机和自助行报警主机使用这种配置延时方式; false, 表示通过
NET_DVR_ALARMSUBSYSTEMPARAM 中的 wEnterDelay、wExitDelay 来设置 -->
<detectorType
  opt="panicButton,magneticContact,smokeDetector,activeInfraredDetector,p
  assiveInfraredDetector,glassBreakDetector,
  vibrationDetector,dualTechnologyPirDetector,tripleTechnologyPirDetector,h
  umidityDetector,temperatureDetector,
  combustibleGasDetecto,dynamicSwitch,controlSwitch,otherDetector" />
<!-- 设备支持的探测器类型, 紧急开关, 门磁开关, 烟感探测器, 主动红外探测器, 被动红外探
测器, 玻璃破碎探测器, 震动探测器, 双鉴移动探测器, 三技术探测器, 温度探测器, 温感探测
器, 可燃气体探测器, 随动开关, 控制开关, 其他探测类型 -->
<zoneType opt="instantZone,
  24hourAudibleZone,delayZone,interiorWithDelayZone,keyswitchZone,supe
  rvisedFireZone,perimeterZone, 24hourSlientZone,disable" />
<!-- req,设备支持的防区类型 -->
<uploadAlarmRecoveryReport>true</uploadAlarmRecoveryReport>
<!-- req,上传报警恢复报告配置 -->
<zoneDelayTime min="30" max="60" />
<!-- req,延时防区延时时间 -->
<sensitivity opt="10ms,250ms,500ms,750ms" />
<!-- req,灵敏度 -->
<arrayBypass>true</arrayBypass>
<!-- req,支持防区组旁路配置-->
<moduleStatus attri="readonly" opt="offline,online" />
<!-- req,模块状态 -->
<moduleAddress min="0" max="0" />
<!-- req,模块地址 -->
<moduleChannel>true</moduleChannel>
<!-- req,模块通道 -->
<moduleType
  opt="localZone,1zoneExpander,2zoneExpander,8ZoneExpander,8sensorZon
  eExpander,1ZoneAndTrigger" />
<!-- 支持的防区类型 -->
<zoneNo attri="readonly" min="1" max="8" />
<!-- req,防区序号, 只能获取 -->
<subsystemNo attri="readonly">1</subsystemNo>
<!-- req,防区所属的子系统号, 只能获取 -->
<alarmType opt="open,close" default="open" />
<!-- req,报警器类型, open-常开, close-常闭 -->
<InDelayTime min="0" max="255" />
<!-- req,进入延迟时间, 单位 s -->
<OutDelayTime min="0" max="255" />

```

```

        <!-- req,退出延迟时间, 单位 s -->
    </ZoneConfig>

    <GetZoneList>
        <enabled>true</enabled>
        <!-- req,支持获取防区列表 -->
    </GetZoneList>

    <ZoneArmDisarm>
        <enabled>true</enabled>
        <!-- req,支持防区布撤防 -->
    </ZoneArmDisarm>

    <ZoneGroupBypass>
        <enabled>true</enabled>
        <!-- req,支持防区组旁路 -->
    </ZoneGroupBypass>
</Zone>

<IOIn>
    <IOInNo attri="readonly" min="1" max="" />
    <!-- req,IO 输入号序号, 只能获取 -->
    <useType opt="disabled,openDoorBtn,magneticStatus,custom" />
    <!-- req,用途: 禁用, 开门按钮, 门状态, 自定义 -->

    <NodeList>
        <Node>
            <id>1</id>
            <name>S1</name>
            <useType opt="disabled,openDoorBtn,magneticStatus,custom" />
        </Node>

        <Node>
            <id>2</id>
            <name>S2</name>
            <useType opt="disabled,openDoorBtn,magneticStatus,custom" />
        </Node>

        <Node>
            <id>3</id>
            <name>S3</name>
            <useType opt="disabled,openDoorBtn,magneticStatus,custom" />
        </Node>
    </NodeList>

```

```

<Node>
  <id>4</id>
  <name>S4</name>
  <useType opt="disabled,openDoorBtn,magneticStatus,custom" />
</Node>

<Node>
  <id>5</id>
  <name>AI1</name>
  <useType opt="disabled,openDoorBtn,magneticStatus,custom" />
</Node>

<Node>
  <id>6</id>
  <name>AI2</name>
  <useType opt="disabled,openDoorBtn,magneticStatus,custom" />
</Node>

<Node>
  <id>7</id>
  <name>AI3</name>
  <useType opt="disabled,openDoorBtn,magneticStatus,custom" />
</Node>

<Node>
  <id>8</id>
  <name>AI4</name>
  <useType opt="disabled,openDoorBtn,magneticStatus,custom" />
</Node>
</NodeList>
</IOIn>

<IOOut>
  <IOOutNo attri="readonly" min="1" max="4" />
  <!-- req,IO 输出号序号，只能获取 -->
  <useType opt="disabled,electricLock,custom" />
  <!-- req,用途：禁用，电锁，自定义 -->
  <NodeList>
    <Node>
      <id>1</id>
      <name>COM1</name>
      <useType opt="disabled,electricLock,custom" />
    </Node>
  </NodeList>

```



```

    <Node>
      <id>2</id>
      <name>COM2</name>
      <useType opt="disabled,electricLock,custom" />
    </Node>

    <Node>
      <id>3</id>
      <name>AO1</name>
      <useType opt="disabled,custom" />
    </Node>

    <Node>
      <id>4</id>
      <name>AO2</name>
      <useType opt="disabled,custom" />
    </Node>
  </NodeList>
</IOOut>

<ElevatorControl>
  <elevatorNo attri="readonly" min="1" max="1" />
  <!-- ro,电梯号 -->
  <interfaceType opt="RS485,network" />
  <!-- req,接口类型: RS485, 网口 -->
  <RS485Protocol opt="private,custom" />
  <!-- req,RS485 协议类型: 私有, 自定义 -->
  <networkProtocol opt="private,custom" />
  <!-- req,网口协议类型: 私有, 自定义 -->
  <serverIP opt="ipv4,ipv6" />
  <!-- opt,梯控服务器 IP -->
  <serverPort min="0" max="65535" />
  <!-- opt,梯控服务器端口号 -->
</ElevatorControl>

<RS485Config>
  <!-- req,485 参数配置, 配置结构体中关于 485 通信的参数能力见"RS232 和 RS485 串口能力节点",其他参数见下, 设备不支持时不返回大节点 -->
  <deviceNameLength min="0" max="32" />
  <!-- req,485 名称 -->
  <deviceType>true</deviceType>
  <!-- req,支持设备类型配置, 具体支持的设备类型通过获取外接设备列表接口获取 -->
  <deviceProtocol>true</deviceProtocol>
  <!-- req,支持设备类型配置, 具体支持的设备协议通过"获取协议列表接口"获取 -->

```

```
</RS485Config>
```

```
<supportVisRegisterInfo opt="true,false" />
```

```
<!-- req,是否支持获取注册信息, 不支持不返回, 且仅有门口机或管理机才可能返回该节点 -->
```

```
<CallRoomConfig>
```

```
<!-- req,别墅机按键呼叫住户配置 -->
```

```
<keyNo min="1" max="4" />
```

```
<!-- req,别墅机按键号 -->
```

```
<floorNo min="1" max="999" />
```

```
<!-- req,别墅层号 -->
```

```
<roomNo min="1" max="999" />
```

```
<!-- req,别墅房间号 -->
```

```
</CallRoomConfig>
```

```
<VideoCall>
```

```
<!-- opt,可视通话能力 -->
```

```
<enabled opt="true,false" />
```

```
<!-- req,是否支持可视通话 -->
```

```
<supportCmd opt="Cancel,Answer,Decline,Timeout,Bye" />
```

```
<!-- req,设备支持的信令类型 Cancel-取消本次呼叫,Answer-接听,Decline-拒接,Timeout-超时,Bye-结束通话 -->
```

```
</VideoCall>
```

```
<VideoIntercomStream>
```

```
<!-- opt,视频源 -->
```

```
<sourceType
```

```
opt="IPC,DVR/DVS/NVR,OutDoorDevice,OutDoorFenceDevice,AgainDevice" />
```

```
<!-- opt,视频源类型 -->
```

```
<againDeviceNumber min="" max="" />
```

```
<!-- opt,二次确认机个数 -->
```

```
</VideoIntercomStream>
```

```
<UploadAlarmCfg>
```

```
<!-- req,NET_DVR_VIDEO_INTERCOM_ALARM_CFG 可视对讲报警事件参数配置 -->
```

```
<UploadDoorNotCloseAlarm opt="true,false" />
```

```
<!-- req,是否上传门未关报警 -->
```

```
</UploadAlarmCfg>
```

```
</VideoIntercom>
```

```
</IpViewDevAbility>
```

7.128 日志类型

dwMajorType

日志主类型

宏定义	宏定义值	含义
MAJOR_ALARM	0x1	报警
MAJOR_EXCEPTION	0x2	异常
MAJOR_OPERATION	0x3	操作
MAJOR_INFORMATION	0x4	日志附加信息

dwMinorType

日志次类型

主类型的宏定义	宏定义值	含义
MAJOR_ALARM	0x1	报警
次类型的宏定义	宏定义值	含义
MINOR_ALARM_IN	0x1	报警输入
MINOR_ALARM_OUT	0x2	报警输出
MINOR_MOTDET_START	0x3	移动侦测报警开始
MINOR_MOTDET_STOP	0x4	移动侦测报警结束
MINOR_HIDE_ALARM_START	0x5	遮挡报警开始
MINOR_HIDE_ALARM_STOP	0x6	遮挡报警结束
MINOR_VCA_ALARM_START	0x7	智能报警开始
MINOR_VCA_ALARM_STOP	0x8	智能报警结束
MINOR_NETALARM_START	0x0b	网络报警开始
MINOR_NETALARM_STOP	0x0c	网络报警结束

主类型的宏定义	宏定义值	含义
MAJOR_EXCEPTION	0x2	异常
次类型的宏定义	宏定义值	含义
MINOR_VI_LOST	0x21	视频信号丢失
MINOR_ILLEGAL_ACCESS	0x22	非法访问
MINOR_HD_FULL	0x23	硬盘满
MINOR_HD_ERROR	0x24	硬盘错误
MINOR_DCD_LOST	0x25	MODEM 掉线(保留)
MINOR_IP_CONFLICT	0x26	IP 地址冲突
MINOR_NET_BROKEN	0x27	网络断开
MINOR_REC_ERROR	0x28	录像出错

MINOR_IPC_NO_LINK	0x29	IPC 连接异常
MINOR_VI_EXCEPTION	0x2a	视频输入异常(只针对模拟通道)
MINOR_IPC_IP_CONFLICT	0x2b	IPC 的 IP 地址冲突
MINOR_SENCE_EXCEPTION	0x2c	场景异常
MINOR_PIC_REC_ERROR	0x2d	抓图出错,获取图片文件失败

主类型的宏定义	宏定义值	含义
MAJOR_OPERATION	0x3	操作
次类型的宏定义	宏定义值	含义
MINOR_START_DVR	0x41	开机
MINOR_STOP_DVR	0x42	关机
MINOR_STOP_ABNORMAL	0x43	异常关机
MINOR_REBOOT_DVR	0x44	本地重启设备
MINOR_LOCAL_LOGIN	0x50	本地登陆
MINOR_LOCAL_LOGOUT	0x51	本地注销登陆
MINOR_LOCAL_CFG_PARM	0x52	本地配置参数
MINOR_LOCAL_PLAYBYFILE	0x53	本地按文件回放或下载
MINOR_LOCAL_PLAYBYTIME	0x54	本地按时间回放或下载
MINOR_LOCAL_START_REC	0x55	本地开始录像
MINOR_LOCAL_STOP_REC	0x56	本地停止录像
MINOR_LOCAL_PTZCTRL	0x57	本地云台控制
MINOR_LOCAL_PREVIEW	0x58	本地预览(保留不使用)
MINOR_LOCAL_MODIFY_TIME	0x59	本地修改时间(保留不使用)
MINOR_LOCAL_UPGRADE	0x5a	本地升级
MINOR_LOCAL_RECFILE_OUTPUT	0x5b	本地备份录象文件
MINOR_LOCAL_FORMAT_HDD	0x5c	本地初始化硬盘
MINOR_LOCAL_CFGFILE_OUTPUT	0x5d	导出本地配置文件
MINOR_LOCAL_CFGFILE_INPUT	0x5e	导入本地配置文件
MINOR_LOCAL_COPYFILE	0x5f	本地备份文件
MINOR_LOCAL_LOCKFILE	0x60	本地锁定录像文件
MINOR_LOCAL_UNLOCKFILE	0x61	本地解锁录像文件
MINOR_LOCAL_DVR_ALARM	0x62	本地手动清除和触发报警
MINOR_IPC_ADD	0x63	本地添加 IPC
MINOR_IPC_DEL	0x64	本地删除 IPC
MINOR_IPC_SET	0x65	本地设置 IPC

MINOR_LOCAL_START_BACKUP	0x66	本地开始备份
MINOR_LOCAL_STOP_BACKUP	0x67	本地停止备份
MINOR_LOCAL_COPYFILE_START_TIME	0x68	本地备份开始时间
MINOR_LOCAL_COPYFILE_END_TIME	0x69	本地备份结束时间
MINOR_LOCAL_ADD_NAS	0x6a	本地添加网络硬盘
MINOR_LOCAL_DEL_NAS	0x6b	本地删除 NAS 盘
MINOR_LOCAL_SET_NAS	0x6c	本地设置 NAS 盘
MINOR_REMOTE_LOGIN	0x70	远程登录
MINOR_REMOTE_LOGOUT	0x71	远程注销登陆
MINOR_REMOTE_START_REC	0x72	远程开始录像
MINOR_REMOTE_STOP_REC	0x73	远程停止录像
MINOR_START_TRANS_CHAN	0x74	开始透明传输
MINOR_STOP_TRANS_CHAN	0x75	停止透明传输
MINOR_REMOTE_GET_PARM	0x76	远程获取参数
MINOR_REMOTE_CFG_PARM	0x77	远程配置参数
MINOR_REMOTE_GET_STATUS	0x78	远程获取状态
MINOR_REMOTE_ARM	0x79	远程布防
MINOR_REMOTE_DISARM	0x7a	远程撤防
MINOR_REMOTE_REBOOT	0x7b	远程重启
MINOR_START_VT	0x7c	开始语音对讲
MINOR_STOP_VT	0x7d	停止语音对讲
MINOR_REMOTE_UPGRADE	0x7e	远程升级
MINOR_REMOTE_PLAYBYFILE	0x7f	远程按文件回放
MINOR_REMOTE_PLAYBYTIME	0x80	远程按时间回放
MINOR_REMOTE_PTZCTRL	0x81	远程云台控制
MINOR_REMOTE_FORMAT_HDD	0x82	远程格式化硬盘
MINOR_REMOTE_STOP	0x83	远程关机
MINOR_REMOTE_LOCKFILE	0x84	远程锁定文件
MINOR_REMOTE_UNLOCKFILE	0x85	远程解锁文件
MINOR_REMOTE_CFGFILE_OUTPUT	0x86	远程导出配置文件
MINOR_REMOTE_CFGFILE_INPUT	0x87	远程导入配置文件
MINOR_REMOTE_RECFILE_OUTPUT	0x88	远程导出录像文件
MINOR_REMOTE_DVR_ALARM	0x89	远程手动清除和触发报警
MINOR_REMOTE_IPC_ADD	0x8a	远程添加 IPC
MINOR_REMOTE_IPC_DEL	0x8b	远程删除 IPC

MINOR_REMOTE_IPC_SET	0x8c	远程设置 IPC
MINOR_REBOOT_VCA_LIB	0x8d	重启智能库
MINOR_REMOTE_ADD_NAS	0x8e	远程添加 NAS 盘
MINOR_REMOTE_DEL_NAS	0x8f	远程删除 NAS 盘
MINOR_REMOTE_SET_NAS	0x90	远程设置 NAS 盘
MINOR_START_DYNAMIC_DECODE	0xb0	开始动态解码
MINOR_STOP_DYNAMIC_DECODE	0xb1	停止动态解码
MINOR_GET_CYC_CFG	0xb2	获取解码器通道轮询配置
MINOR_SET_CYC_CFG	0xb3	设置解码通道轮询配置
MINOR_START_CYC_DECODE	0xb4	开始轮询解码
MINOR_STOP_CYC_DECODE	0xb5	停止轮询解码
MINOR_GET_DECCHAN_STATUS	0xb6	获取解码通道状态
MINOR_GET_DECCHAN_INFO	0xb7	获取解码通道当前信息
MINOR_START_PASSIVE_DEC	0xb8	开始被动解码
MINOR_STOP_PASSIVE_DEC	0xb9	停止被动解码
MINOR_CTRL_PASSIVE_DEC	0xba	控制被动解码
MINOR_RECON_PASSIVE_DEC	0xbb	被动解码重连
MINOR_GET_DEC_CHAN_SW	0xbc	获取解码通道总开关
MINOR_SET_DEC_CHAN_SW	0xbd	设置解码通道总开关
MINOR_CTRL_DEC_CHAN_SCALE	0xbe	解码通道缩放控制
MINOR_SET_REMOTE_REPLAY	0xbf	设置远程回放
MINOR_GET_REMOTE_REPLAY	0xc0	获取远程回放状态
MINOR_CTRL_REMOTE_REPLAY	0xc1	远程回放控制
MINOR_SET_DISP_CFG	0xc2	设置显示通道
MINOR_GET_DISP_CFG	0xc3	获取显示通道设置
MINOR_SET_PLANTABLE	0xc4	设置计划轮询表
MINOR_GET_PLANTABLE	0xc5	获取计划轮询表
MINOR_START_PPPOE	0xc6	开始 PPPoE 连接
MINOR_STOP_PPPOE	0xc7	结束 PPPoE 连接
MINOR_UPLOAD_LOGO	0xc8	上传 LOGO
MINOR_DISPLAY_LOGO	0x205	显示 LOGO
MINOR_HIDE_LOGO	0x206	隐藏 LOGO
MINOR_SET_DEC_DELAY_LEVEL	0x207	解码通道延时级别设置
MINOR_SET_BIGSCREEN_DIPLAY_AREA	0x208	设置大屏显示区域
MINOR_CUT_VIDEO_SOURCE	0x209	大屏视频源切割设置

MINOR_SET_BASEMAP_AREA	0x210	大屏底图区域设置
MINOR_DOWNLOAD_BASEMAP	0x211	下载大屏底图
MINOR_CUT_BASEMAP	0x212	底图切割配置
MINOR_CONTROL_ELEC_ENLARGE	0x213	电子放大操作（放大或还原）
MINOR_SET_OUTPUT_RESOLUTION	0x214	显示输出分辨率设置
MINOR_SET_TRANSPARENCY	0x215	图层透明度设置
MINOR_SET_OSD	0x216	显示 OSD 设置
MINOR_RESTORE_DEC_STATUS	0x217	恢复初始状态（场景切换时，解码恢复初始状态）
MINOR_SCREEN_SET_INPUT	0x251	修改输入源
MINOR_SCREEN_SET_DISPLAY	0x252	修改输出通道
MINOR_SCREEN_SET_OSD	0x253	修改虚拟 LED
MINOR_SCREEN_SET_LOGO	0x254	修改 LOGO
MINOR_SCREEN_SET_LAYOUT	0x255	设置布局
MINOR_SCREEN_PICTUREPREVIEW	0x256	回显操作
MINOR_SCREEN_GET_OSD	0x257	获取虚拟 LED
MINOR_SCREEN_GET_LAYOUT	0x258	获取场景
MINOR_SCREEN_LAYOUT_CTRL	0x259	场景控制
MINOR_GET_ALL_VALID_WND	0x260	获取所有有效窗口
MINOR_GET_SIGNAL_WND	0x261	获取单个窗口信息
MINOR_WINDOW_CTRL	0x262	窗口控制
MINOR_GET_LAYOUT_LIST	0x263	获取场景列表
MINOR_LAYOUT_CTRL	0x264	场景控制
MINOR_SET_LAYOUT	0x265	设置单个场景
MINOR_GET_SIGNAL_LIST	0x266	获取输入信号源列表
MINOR_GET_PLAN_LIST	0x267	获取预案列表
MINOR_SET_PLAN	0x268	修改预案
MINOR_CTRL_PLAN	0x269	控制预案
MINOR_CTRL_SCREEN	0x270	屏幕控制
MINOR_ADD_NETSIG	0x271	添加信号源
MINOR_SET_NETSIG	0x272	修改信号源
MINOR_SET_DECBDCFG	0x273	设置解码板参数
MINOR_GET_DECBDCFG	0x274	获取解码板参数
MINOR_GET_DEVICE_STATUS	0x275	获取设备信息
MINOR_UPLOAD_PICTURE	0x276	底图上传
MINOR_SET_USERPWD	0x277	设置用户密码

MINOR_ADD_LAYOUT	0x278	添加场景
MINOR_DEL_LAYOUT	0x279	删除场景
MINOR_DEL_NETSIG	0x280	删除信号源
MINOR_ADD_PLAN	0x281	添加预案
MINOR_DEL_PLAN	0x282	删除预案
MINOR_GET_EXTERNAL_MATRIX_CFG	0x283	获取外接矩阵配置
MINOR_SET_EXTERNAL_MATRIX_CFG	0x284	设置外接矩阵配置
MINOR_GET_USER_CFG	0x285	获取用户配置
MINOR_SET_USER_CFG	0x286	设置用户配置
MINOR_GET_DISPLAY_PANEL_LINK_CFG	0x287	获取显示墙连接配置
MINOR_SET_DISPLAY_PANEL_LINK_CFG	0x288	设置显示墙连接配置
MINOR_GET_WALLSCENE_PARAM	0x289	获取电视墙场景
MINOR_SET_WALLSCENE_PARAM	0x28a	设置电视墙场景
MINOR_GET_CURRENT_WALLSCENE	0x28b	获取当前使用场景
MINOR_SWITCH_WALLSCENE	0x28c	场景切换

主类型的宏定义	宏定义值	含义
MAJOR_INFORMATION	0x4	附加信息
次类型的宏定义	宏定义值	含义
MINOR_HDD_INFO	0xa1	硬盘信息
MINOR_SMART_INFO	0xa2	S.M.A.R.T 信息
MINOR_REC_START	0xa3	开始录像
MINOR_REC_STOP	0xa4	停止录像
MINOR_REC_OVERDUE	0xa5	过期录像删除
MINOR_LINK_START	0xa6	连接前端设备
MINOR_LINK_STOP	0xa7	断开前端设备
MINOR_NET_DISK_INFO	0xa8	网络硬盘信息