

设备(拼控)

网络 SDK 编程指南

V5.2

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	12
3.1 解码器主要流程	12
3.2 视频综合平台主要流程	13
3.3 多屏控制器主要流程	15
3.4 电视墙功能接口流程	17
3.5 报警模块流程	19
3.5.1 报警布防	19
3.5.2 报警监听	20
3.6 电视墙号、窗口号等编号说明	21
4 函数调用实例	22
4.1 解码上墙示例代码	22
5 功能接口介绍	28
5.1 通用接口介绍	28
5.2 多路解码器	28
5.3 视频综合平台	33
5.4 集中式多屏控制器	42
5.5 分布式多屏控制器	48
6 函数说明	53
6.1 SDK 初始化	53
6.1.1 初始化 SDK NET_DVR_Init	53
6.1.2 释放 SDK 资源 NET_DVR_Cleanup	53
6.2 SDK 本地功能	53
SDK 本地参数配置	53
6.2.1 获取 SDK 本地参数 NET_DVR_GetSDKLocalCfg	53
6.2.2 设置 SDK 本地参数 NET_DVR_SetSDKLocalCfg	54
连接和接收超时时间及重连设置	55
6.2.3 设置网络连接超时时间和连接尝试次数 NET_DVR_SetConnectTime	55
6.2.4 设置重连功能 NET_DVR_SetReconnect	55
6.2.5 设置接收超时时间 NET_DVR_SetRecvTimeOut	55
多网卡绑定	55
6.2.6 获取所有 IP，用于支持多网卡接口 NET_DVR_GetLocalIP	55
6.2.7 设置 IP 绑定 NET_DVR_SetValidIP	56
SDK 版本、状态和能力	56
6.2.8 获取 SDK 的版本号和 build 信息 NET_DVR_GetSDKBuildVersion	56
6.2.9 获取当前 SDK 的状态信息 NET_DVR_GetSDKState	56
6.2.10 获取当前 SDK 的功能信息 NET_DVR_GetSDKAbility	56
SDK 启用写日志	57
6.2.11 启用写日志文件 NET_DVR_SetLogToFile	57

异常消息回调	57
6.2.12 设置异常消息回调函数 NET_DVR_SetExceptionCallBack_V30	57
获取错误信息	59
6.2.13 返回最后操作的错误码 NET_DVR_GetLastError	59
6.2.14 返回最后操作的错误码信息 NET_DVR_GetErrorMsg	59
6.3 用户注册	60
6.3.1 激活设备 NET_DVR_ActivateDevice	60
6.3.2 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	60
6.3.3 用户注册设备 NET_DVR_Login_V40	60
6.3.4 用户注销 NET_DVR_Logout	61
6.4 设备能力集	61
6.4.1 获取设备能力集 NET_DVR_GetDeviceAbility	61
6.5 布防、撤防	63
设置报警等信息上传的回调函数	63
6.5.1 注册报警信息回调函数 NET_DVR_SetDVRMessageCallBack_V30	63
布防撤防	63
6.5.2 报警布防，建立报警上传通道 NET_DVR_SetupAlarmChan_V41	63
6.5.3 撤销报警上传通道 NET_DVR_CloseAlarmChan_V30	64
6.6 监听报警	64
6.6.1 启动监听，接收设备主动上传的报警等信息 NET_DVR_StartListen_V30	64
6.6.2 停止监听（支持多线程）NET_DVR_StopListen_V30	65
6.7 通用参数配置	65
6.7.1 获取设备的配置信息 NET_DVR_GetDVRConfig	65
6.7.2 设置设备的配置信息 NET_DVR_SetDVRConfig	66
6.7.3 批量获取设备参数 NET_DVR_GetDeviceConfig	67
6.7.4 批量设置设备参数 NET_DVR_SetDeviceConfig	68
6.8 信号源和设备管理	69
视频综合平台子系统信息	69
6.8.1 获取所有子系统信息 NET_DVR_MatrixGetSubSystemInfo_V40	69
6.8.2 设置子系统参数 NET_DVR_MatrixSetSubSystemInfo_V40	69
信号源管理	69
6.8.3 获取信号源列表 NET_DVR_GetInputSignalList_V40	69
6.8.4 设置信号源参数 NET_DVR_SetDVRConfig	70
6.8.5 批量获取信号源参数 NET_DVR_GetDeviceConfig	70
6.8.6 批量设置信号源参数 NET_DVR_SetDeviceConfig	71
子板和子板异常信息	72
6.8.7 获取子板和子板异常信息 NET_DVR_GetDVRConfig	72
分布式多屏控制器子设备	72
6.8.8 获取分布式子设备信息列表 NET_DVR_GetDevList	72
6.8.9 批量获取子设备信息 NET_DVR_GetDeviceConfig	73
6.8.10 批量设置子设备信息 NET_DVR_SetDeviceConfig	74
屏幕服务器登录参数配置	74
6.8.11 屏幕服务器登录参数配置 NET_DVR_STDXMLConfig	74
输入板参数配置	75

6.8.12	获取输入板参数 NET_DVR_GetSTDConfig	75
6.8.13	设置输入板参数 NET_DVR_SetSTDConfig.....	76
	显示输入配置	76
6.8.14	获取显示输入参数能力集 NET_DVR_GetSTDAbility	76
6.8.15	获取显示输入参数 NET_DVR_GetSTDConfig	77
6.8.16	设置显示输入参数 NET_DVR_SetSTDConfig.....	77
	虚拟屏配置	78
6.8.17	获取虚拟屏参数 NET_DVR_GetSTDConfig	78
6.8.18	设置虚拟屏参数 NET_DVR_SetSTDConfig.....	78
6.8.19	获取虚拟屏参数 NET_DVR_GetDVRConfig	79
6.8.20	设置虚拟屏参数 NET_DVR_SetDVRConfig	79
6.9	电视墙操作.....	80
6.9.1	获取电视墙相关参数 NET_DVR_GetDVRConfig.....	80
6.9.2	设置电视墙相关参数 NET_DVR_SetDVRConfig	80
6.9.3	批量获取显示输出信息 NET_DVR_GetDeviceConfig.....	81
6.9.4	批量设置显示输出参数 NET_DVR_SetDeviceConfig	82
6.9.5	电视墙远程控制 NET_DVR_RemoteControl.....	82
6.9.6	获取 LED 区域信息 NET_DVR_GetSTDConfig	83
6.10	电视墙窗口操作.....	83
6.10.1	获取窗口参数 NET_DVR_GetDVRConfig	83
6.10.2	设置窗口参数 NET_DVR_SetDVRConfig	84
6.10.3	批量获取窗口信息 NET_DVR_GetDeviceConfig.....	84
6.10.4	批量设置窗口 NET_DVR_SetDeviceConfigEx	85
6.10.5	获取窗口参数配置 NET_DVR_MatrixGetDecChanCfg.....	86
6.10.6	设置窗口参数配置 NET_DVR_MatrixSetDecChanCfg	86
6.10.7	远程控制 NET_DVR_RemoteControl.....	87
6.11	窗口解码操作.....	87
	主动解码	87
6.11.1	启动动态解码 NET_DVR_MatrixStartDynamic_V41.....	87
6.11.2	停止动态解码 NET_DVR_MatrixStopDynamic	87
6.11.3	批量设置动态解码 NET_DVR_SetDeviceConfig	88
6.11.4	获取轮巡解码通道 NET_DVR_MatrixGetLoopDecChanInfo_V41	88
6.11.5	设置轮巡解码通道 NET_DVR_MatrixSetLoopDecChanInfo_V41	89
6.11.6	远程文件回放解码 NET_DVR_RemoteControl	89
6.11.7	远程回放文件控制 NET_DVR_MatrixSetRemotePlayControl.....	89
6.11.8	获取回放状态 NET_DVR_MatrixGetRemotePlayStatus	90
	解码开关控制	90
6.11.9	获取窗口解码开关 NET_DVR_MatrixGetDecChanEnable.....	90
6.11.10	设置窗口解码开关 NET_DVR_MatrixSetDecChanEnable	91
6.11.11	批量获取解码通道开关 NET_DVR_GetDeviceConfig.....	91
6.11.12	批量设置解码通道开关 NET_DVR_SetDeviceConfig	92
6.11.13	获取窗口轮巡解码开关 NET_DVR_MatrixGetLoopDecChanEnable	93
6.11.14	设置窗口轮巡解码开关 NET_DVR_MatrixSetLoopDecChanEnable.....	93
	被动解码	93

6.11.15	启动被动解码 NET_DVR_MatrixStartPassiveDecode	93
6.11.16	向被动解码窗口发送数据 NET_DVR_MatrixSendData	93
6.11.17	停止被动解码 NET_DVR_MatrixStopPassiveDecode.....	94
6.11.18	获取被动解码状态 NET_DVR_MatrixGetPassiveDecodeStatus	94
6.11.19	被动解码播放控制 NET_DVR_MatrixPassiveDecodeControl	94
	窗口解码信息	94
6.11.20	获取当前窗口解码信息 NET_DVR_MatrixGetDecChanInfo_V41	94
6.11.21	批量获取窗口解码信息 NET_DVR_GetDeviceConfig.....	95
6.11.22	获取窗口解码状态信息 NET_DVR_GetDeviceStatus	96
	电子放大	96
6.11.23	大屏电子放大 NET_DVR_ScreenZoomIn	96
	JPEG 抓图	97
6.11.24	单帧数据捕获并保存成 JPEG 图片 NET_DVR_CaptureJPEGPicture	97
6.11.25	单帧数据捕获并保存成 JPEG 存放在指定的内存空间中 NET_DVR_CaptureJPEGPicture_NEW.....	97
6.12	音频切换.....	97
6.12.1	批量获取音频切换参数 NET_DVR_GetDeviceConfig.....	97
6.12.2	批量设置音频切换参数 NET_DVR_SetDeviceConfig	98
6.13	场景功能.....	99
	配置和控制	99
6.13.1	批量获取场景参数 NET_DVR_GetDeviceConfig.....	99
6.13.2	批量设置场景参数 NET_DVR_SetDeviceConfig	100
6.13.3	场景切换控制 NET_DVR_MatrixSceneControl	100
6.13.4	远程控制场景 NET_DVR_RemoteControl.....	101
6.13.5	获取当前正在使用的场景模式 NET_DVR_MatrixGetCurrentSceneMode	101
	场景配置文件导入导出	101
6.13.6	导入场景配置文件 NET_DVR_UploadFile	101
6.13.7	获取文件导入的进度和状态 NET_DVR_GetUploadState	102
6.13.8	停止导入配置文件 NET_DVR_UploadClose	102
6.13.9	导出场景配置文件 NET_DVR_StartDownload	102
6.13.10	获取文件导出的进度和状态 NET_DVR_GetDownloadState	103
6.13.11	停止导出配置文件 NET_DVR_StopDownload.....	103
6.14	预案功能.....	103
6.14.1	获取预案列表 NET_DVR_GetPlanList	103
6.14.2	远程控制预案 NET_DVR_RemoteControl.....	104
6.14.3	获取预案信息 NET_DVR_GetDVRConfig	104
6.14.4	设置预案 NET_DVR_SetDVRConfig	104
6.15	底图配置、上传和控制	105
6.15.1	获取底图参数 NET_DVR_GetDVRConfig	105
6.15.2	设置底图参数 NET_DVR_SetDVRConfig	105
6.15.3	图片上传 NET_DVR_PicUpload	106
6.15.4	获取图片上传的进度 NET_DVR_GetPicUploadProgress	106
6.15.5	获取图片上传的状态 NET_DVR_GetPicUploadState	106
6.15.6	关闭图片上传 NET_DVR_CloseUploadHandle.....	107
6.15.7	删除底图 NET_DVR_RemoteControl.....	107

6.16	LOGO 配置、上传和控制.....	107
6.16.1	获取 LOGO 参数 NET_DVR_GetDVRConfig	107
6.16.2	设置 LOGO 参数 NET_DVR_SetDVRConfig.....	108
6.16.3	LOGO 上传 NET_DVR_UploadLogo_NEW.....	108
6.16.4	LOGO 下载 NET_DVR_DownloadLogo.....	108
6.16.5	删除 LOGO NET_DVR_RemoteControl	109
6.17	解码 OSD 配置.....	109
6.17.1	获取解码 OSD 能力集 NET_DVR_GetSTDAbility	109
6.17.2	获取解码 OSD 参数 NET_DVR_GetSTDConfig	110
6.17.3	设置解码 OSD 参数 NET_DVR_SetSTDConfig.....	110
6.18	图片回显.....	111
6.18.1	开始图片回显 NET_DVR_StartPicPreview	111
6.18.2	停止图片回显 NET_DVR_StopScreenPic	111
6.19	图像微调.....	112
6.19.1	图像微调 NET_DVR_MatrixPicAdjust.....	112
6.20	透明通道.....	112
6.20.1	建立透明通道 NET_DVR_SerialStart.....	112
6.20.2	通过透明通道向设备串口发送数据 NET_DVR_SerialSend.....	113
6.20.3	断开透明通道 NET_DVR_SerialStop	113
6.21	云台控制.....	113
6.21.1	云台控制 NET_DVR_PTZControlWithSpeed_Other	113
6.21.2	3 维带速度云台控制 NET_DVR_RemoteControl.....	115
6.21.3	获取设备支持的云台协议列表 NET_DVR_GetPTZProtocol_Ex.....	116
6.21.4	获取云台控制参数 NET_DVR_GetDVRConfig	116
6.21.5	设置云台控制参数 NET_DVR_SetDVRConfig	116
6.22	相机拼接配置.....	117
6.22.1	获取流来源信息 NET_DVR_GetDeviceConfig	117
6.22.2	设置流来源信息 NET_DVR_SetDeviceConfig	118
6.22.3	获取相机拼接参数 NET_DVR_GetDVRConfig	119
6.22.4	设置相机拼接参数 NET_DVR_SetDVRConfig	119
6.23	信号源拼接裁剪配置.....	120
6.23.1	获取拼接裁剪参数 NET_DVR_GetDVRConfig	120
6.23.2	批量获取拼接参数 NET_DVR_GetDeviceConfig.....	120
6.23.3	批量设置拼接参数 NET_DVR_SetDeviceConfigEx.....	121
6.23.4	拼接裁剪控制 NET_DVR_RemoteControl.....	122
6.24	光端机相关配置.....	122
6.24.1	获取光信号相关参数 NET_DVR_GetDVRConfig.....	122
6.24.2	设置光信号相关参数 NET_DVR_SetDVRConfig	123
6.25	获取 IPC 协议列表.....	124
6.25.1	获取设备支持的 IPC 协议列表 NET_DVR_GetIPCProtoList_V41	124
6.26	获取设备状态.....	124
6.26.1	获取设备状态信息 NET_DVR_GetDeviceStatus	124
6.26.2	设备在线状态检测 NET_DVR_RemoteControl	125
6.26.3	获取视频综合平台状态 NET_DVR_GetDVRConfig.....	125

6.27	EDID 文件管理.....	126
	EDID 文件上传	126
6.27.1	上传文件 NET_DVR_UploadFile_V40.....	126
6.27.2	获取文件上传的进度和状态 NET_DVR_GetUploadState	126
6.27.3	停止文件上传 NET_DVR_UploadClose	126
	EDID 文件信息配置	127
6.27.4	获取 EDID 文件信息配置能力集 NET_DVR_GetSTDAbility.....	127
6.27.5	获取 EDID 文件信息 NET_DVR_GetSTDConfig	127
6.27.6	设置 EDID 文件信息 NET_DVR_SetSTDConfig	127
6.27.7	删除 EDID 文件信息 NET_DVR_STDCControl	128
6.28	设备维护.....	128
	远程升级	128
6.28.1	设置远程升级时网络环境 NET_DVR_SetNetworkEnvironment.....	128
6.28.2	远程升级 NET_DVR_Upgrade	129
6.28.3	获取远程升级的进度 NET_DVR_GetUpgradeProgress	129
6.28.4	获取远程升级的状态 NET_DVR_GetUpgradeState	129
6.28.5	获取远程升级的阶段信息 NET_DVR_GetUpgradeStep.....	129
6.28.6	关闭远程升级句柄，释放资源 NET_DVR_CloseUpgradeHandle	130
6.28.7	开始升级子设备 NET_DVR_StartUpgrade.....	130
6.28.8	停止升级子设备 NET_DVR_StopUpgrade	131
	日志查找	131
6.28.9	查找设备的日志信息 NET_DVR_FindDVRLog_V30.....	131
6.28.10	逐条获取查找到的日志信息 NET_DVR_FindNextLog_V30	131
6.28.11	查找视频综合平台的日志信息 NET_DVR_FindDVRLog_Matrix.....	132
6.28.12	逐条获取查找到的平台日志信息 NET_DVR_FindNextLog_MATRIX	132
6.28.13	释放查找日志的资源 NET_DVR_FindLogClose_V30.....	133
	远程恢复默认参数、重启	133
6.28.14	恢复设备默认参数 NET_DVR_RestoreConfig.....	133
6.28.15	重启设备 NET_DVR_RebootDVR.....	133
6.28.16	恢复默认参数、重启 NET_DVR_SetDeviceConfig	133
	配置文件导入导出方式一	134
6.28.17	导出配置文件 NET_DVR_GetConfigFile	134
6.28.18	导入配置文件 NET_DVR_SetConfigFile	135
6.28.19	导出视频综合平台配置文件 NET_DVR_MatrixGetConfigFile	135
6.28.20	导入视频综合平台配置文件 NET_DVR_MatrixSetConfigFile.....	135
	配置文件导入导出方式二	136
6.28.21	上传文件 NET_DVR_UploadFile_V40.....	136
6.28.22	获取文件上传的进度和状态 NET_DVR_GetUploadState	136
6.28.23	停止文件上传 NET_DVR_UploadClose	136
6.28.24	开始下载文件 NET_DVR_StartDownload	137
6.28.25	获取文件下载的进度和状态 NET_DVR_GetDownloadState	137
6.28.26	停止文件下载 NET_DVR_StopDownload.....	137
7	错误代码及说明	138
7.1	网络通讯库错误码	138

7.2	RTSP 通讯库错误码.....	143
7.3	软解码库错误码.....	144
8	结构体说明	145

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、视频综合平台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。本文档主要介绍多屏控制器、解码器、视频综合平台等拼控设备电视墙相关的功能。

本文档中并未描述所有的接口，如预览、回放、通用配置（命令）等接口，对于这些接口的调用，需要查看《设备网络 SDK 使用手册.chm》。

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部分自行解码将更灵活，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。对于拼控设备，不需要使用该库文件。**

2 SDK 版本更新

Version 5.2.5.40 (build20161102)

- DS-B21 V1.3
- 新增显示输入参数能力集（对应接口：[NET_DVR_GetSTDAbility](#)）：
命令：NET_DVR_GET_DISPINPUT_CFG_CAP。
- 新增显示输入参数配置功能（对应接口：[NET_DVR_GetSTDConfig](#)、[NET_DVR_SetSTDConfig](#)）：
命令：NET_DVR_GET_DISPINPUT_CFG、NET_DVR_GET_DISPINPUT_CFG_LIST、NET_DVR_SET_DISPINPUT_CFG。
- [NET_DVR_INPUTSTREAMCFG_V40](#)(输入源参数)扩展：
 - 1) 使用 4 个字节新增参数 dwInputSignalNo(输入信号源编号)，兼容 wInputNo。
 - 2) [NET_DVR_CAM_MODE](#)(信号输入源类型)新增枚举类型：NET_DVR_CAM_DVIT(DVI 双链路)、NET_DVR_CAM_FUSION(多 IPC 拼接)、NET_DVR_CAM_VSCREEN(虚拟屏)、NET_DVR_CAM_FIBER(光纤)、NET_DVR_CAM_3GSDI(3G-SDI)。
- [NET_DVR_WALLOUTPUTPARAM](#)(电视墙显示输出参数)使用 1 个保留字节新增参数：
byUseEDIDResolution(是否使用 EDID 分辨率)。
- [NET_DVR_SUBSYSTEMINFO_V40](#)(子系统信息)中 byInterfaceType(接口类型)新增取值：13-HDBaseT、14-DP、15-DVIT。
- [NET_DVR_PLAN_INFO](#)(预案信息)扩展：
 - 1) 使用 8 个保留字节新增参数：dwBaseMapWndNo(底图窗口号)、dwBaseMapNo(底图号)。
 - 2) 参数 byType(预案执行类型)新增枚举类型取值：NET_DVR_SWITCH_BASEMAP(底图切换)。
- 设备软硬件能力集(BasicCapability，对应接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：DEVICE_SOFTWARE_ABILITY)扩展，<SoftwareCapability>(软件能力)中新增子节点：
<isNotSupportSummerTime>(是否不支持夏令时)。
- RS232 和 RS485 串口能力(SerialAbility，对应接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：DEVICE_SERIAL_ABILITY)扩展：
 - 1) plnBuf 对应的串口能力获取输入描述 XML 增加节点：<subBoardNo>(子板号)。
 - 2) <SerialProtocol>(支持的串口协议)新增子节点：<ScreenCtrlProtocol>(支持的屏幕控制协议)、<MatrixCtrlProtocol>(支持的矩阵控制协议)。
- 电视墙能力集([WallAbility](#)，对应接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：WALL_ABILITY)扩展：
 - 1) 新增节点：<isSupportBaseMapCycleSwitch>(预案是否支持底图切换)、<isSupportEDIDResolution>(是否支持 EDID 分辨率)。
 - 2) <VoutResource>(视频输出资源能力)新增子节点：<SupportLEDOutputConfigAll>(LED 输出口参数是否支持全墙配置)、<supportLEDResolutionVoutType>(支持 LED 分辨率的输出口类型)。
- 新增支持设备类型：DS_B21_MCU_NP_CLASS、DS_B21_S10_x_CLASS、DS_6402HFH_B21D_CLASS、DS_6508HD_B21D_CLASS、DS_iVMSE_B21HW_CLASS。
- 新增错误码：974~991。

Version 5.2.5.15 (build20160906)

- DS-6900UD V2.0
- 新增解码 OSD 配置能力集（对应接口：[NET_DVR_GetSTDAbility](#)）：

命令: NET_DVR_GET_SUBWND_DECODE_OSD_CAP。

- 新增子窗口解码 OSD 配置功能（对应接口: [NET_DVR_GetSTDConfig](#)、[NET_DVR_SetSTDConfig](#)）：
命令: NET_DVR_GET_SUBWND_DECODE_OSD_ALL、NET_DVR_GET_SUBWND_DECODE_OSD、NET_DVR_SET_SUBWND_DECODE_OSD。
- 新增已开会的会议信息报警功能（对应接口: [NET_DVR_SetDVRMessageCallBack_V30](#)）：
命令: COMM_ALARM_WALL_CONFERNECE。
- 新增获取 LED 区域列表的功能（对应接口: [NET_DVR_GetSTDConfig](#)）：
命令: NET_DVR_GET_LED_AREA_INFO_LIST。
- [NET_DVR_WALL_CFG](#)(电视墙全局参数)使用 9 个字节新增参数: byStreamFailedMode(取流失败显示模式)、struSubStreamSwitch(子码流切换)。
- [NET_DVR_VIDEOWALLWINDOWPOSITION](#)(电视墙窗口信息)使用 8 个保留字节新增参数: dwXCoordinate(LED 区域左上角 X 坐标)、dwYCoordinate(LED 区域左上角 Y 坐标)。
- [NET_DVR_DEV_CHAN_INFO_EX](#)(前端编码设备信息)中 byChanType(通道类型)新增取值: 6-屏幕服务器。
- 电视墙能力集([WallAbility](#), 对应接口: [NET_DVR_GetDeviceAbility](#), 能力集类型: WALL_ABILITY)扩展:
 - 1) 新增节点: <isSupportLEDArea>(是否支持 LED 区域)、<SubStreamAutoSwitchCap>(子码流自动切换)、<streamFailedMode>(窗口取流失败显示模式)、<isSupportWallConference>(是否支持会议电视墙)。
 - 2) <WindowCap>(窗口能力)中新增子节点: <wndWidthAlignUnit>(物理分辨率窗口宽度对齐单元)、<wndHeightAlignUnit>(物理分辨率窗口高度对齐单元)、<isSupportDecodeOSD>(是否支持解码 OSD 配置)。

Version 5.1.6.15 (build20160302)

- DS-C10S V2.1
- 新增屏幕服务器登录参数配置功能，对应接口: [NET_DVR_STDXMLConfig](#)。
- 新增输入板参数配置功能（对应接口: [NET_DVR_GetSTDConfig](#)、[NET_DVR_SetSTDConfig](#)）：
命令: NET_DVR_GET_INPUT_BOARD_CFG_LIST、NET_DVR_GET_INPUT_BOARD_CFG、NET_DVR_SET_INPUT_BOARD_CFG。
- 新增解除电视墙输出接口绑定控制命令（对应接口: [NET_DVR_RemoteControl](#)）：
NET_DVR_RESET_VIDEOWALLDISPLAYPOSITION。
- [NET_DVR_INPUTSTREAMCFG_V40](#)(输入源参数)使用 1 个保留字节新增参数: byScreenServer(关联屏幕服务器)。
- [NET_DVR_CAM_MODE](#)(输入源类型)增加枚举类型: NET_DVR_CAM_HDBASET。
- [NET_DVR_VIDEOWALLWINDOWPOSITION](#)(电视墙窗口信息)使用 21 个保留字节新增参数: byWndOperateMode(窗口操作模式)、struResolution(目的窗口分辨率坐标)。
- [NET_DVR_WALL_WIN_STATUS](#)(窗口状态信息)中参数 byStreamType(编码类型)新增取值: NET_DVR_ENCODER_H265(标准 H.265)。
- [NET_DVR_START_PIC_VIEW_INFO](#)(图片回显参数)使用 1 个保留字节新增参数: bySupportStreamView(客户端是否支持码流回显)。
- [NET_DVR_SUBBOARD_INFO](#)(子板信息)中 byBoardType 新增取值: 14-HDBaseT 输入。
- 电视墙能力集([WALL_ABILITY](#))扩展，新增节点: <isSupportPhysicsResolutionWnd>(是否支持物理分辨率开窗)、<ResetOutputDisplayPosition>(是否支持电视墙输出通道解绑)、<InputBoardCfgList>(输入板参数配置能力)。
- 新增错误码: 971、972、973。

Version 5.1.3.25 (build20150916)

- DS-B20 V1.4

- 新增支持信号源拼接功能并且新增删除拼接信号源的功能（对应接口：[NET_DVR_RemoteControl](#)）：
命令：NET_DVR_DEL_SIGNAL_JOINT。
- 新增光纤板环网配置功能（对应接口：[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)）：
命令：NET_DVR_GET_NS_RING_CFG、NET_DVR_SET_NS_RING_CFG。
- 新增光纤板环网状态获取功能（对应接口：[NET_DVR_GetDVRConfig](#)）：
命令：NET_DVR_GET_NS_RING_STATUS。
- 新增光口信息配置功能（对应接口：[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)）：
命令：NET_DVR_GET_OPTICAL_PORT_INFO、NET_DVR_SET_OPTICAL_PORT_INFO。
- 新增编码通道关联光口输入源参数配置功能（对应接口：[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)）：
命令：NET_DVR_GET_OPTICAL_CHAN_RELATE_CFG、NET_DVR_SET_OPTICAL_CHAN_RELATE_CFG。
- 新增支持：批量动态解码、批量解码通道开关配置功能、屏幕控制功能以及获取正在执行预案信息的功能。
- [NET_DVR_SUBSYSTEMINFO_V40](#)(子系统信息)中 bySubSystemType(子系统类型)新增取值：17-X86 拼接子系统、18-普通拼接子系统。
- [NET_DVR_SUBSYSTEM_STATUS](#)(子系统状态)中 bySubSystemType(子系统类型)新增取值：15-多相机拼接子系统、16-网络光纤交换子系统、17-X86 拼接子系统、18-普通拼接子系统。
- [NET_DVR_PLAN_INFO](#)(预案信息)使用 4 个保留字节新增参数：dwSerialNo(串口号)。
- [NET_DVR_GBT28181_ACCESS_CFG](#)(GBT28181 协议接入配置)使用 1 个保留字节新增参数：
byTransProtocol(传输协议)。
- 电视墙能力集([WallAbility](#)，对应接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：WALL_ABILITY)扩展：
 - 1) 新增节点：<SupportBatchDec>(支持批量动态解码和批量解码开关)。
 - 2) <JointSignal>(拼接信号源能力)新增子节点：<JointScale>(拼接规模能力)、<DeleteJoint>(是否支持删除拼接信号源)。
- 设备通用能力集(DEVICE_ABILITY_INFO)，其中 GB/T28181 能力(GBT28181AccessAbility)，新增节点：
<protocolType>(支持的协议类型)。
- 新增能力集类型（对应接口：[NET_DVR_GetDeviceAbility](#)）：
OPTICAL_DEV_ACCESS_ABILITY(光端机接入能力集)、
NET_RING_ABILITY(环网能力集)。

Version 5.1.3.6 (build20150708)

- DS-C10S V2.0
- 获取信号源列表接口扩展：[NET_DVR_GetInputSignalList_V40](#)，向下兼容 NET_DVR_GetInputSignalList。
- 信号源设置命令扩展（对应接口：[NET_DVR_SetDVRConfig](#)）：
命令：NET_DVR_SET_INPUTSTREAMCFG_V40，向下兼容 NET_DVR_SET_INPUTSTREAMCFG。
- 新增信号源拼接功能（对应接口：[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDeviceConfigEx](#)、[NET_DVR_GetDeviceConfig](#)）：
命令：NET_DVR_GET_SIGNAL_JOINT、NET_DVR_SET_SIGNAL_JOINT、NET_DVR_GET_ALL_SIGNAL_JOINT。
- 新增信号源裁剪功能（对应接口：[NET_DVR_RemoteControl](#)、[NET_DVR_GetDVRConfig](#)）：
命令：NET_DVR_SIGNAL_CUT、NET_DVR_SIGNAL_CUTPARAM_GET。
- 新增批量动态解码功能（对应接口：[NET_DVR_SetDeviceConfig](#)）：
命令：NET_DVR_DYNAMIC_DECODE_BATCH。
- 新增批量解码通道开关控制功能（对应接口：[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDeviceConfig](#)）：
命令：NET_DVR_DECSWITCH_GET_BATCH、NET_DVR_DECSWITCH_SET_BATCH。

- 新增获取正在执行预案信息的功能（对应接口：[NET_DVR_GetDVRConfig](#)）：
NET_DVR_GET_PLAYING_PLAN。
- 新增输入源反向云台控制配置功能（对应接口：[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)）：
命令：NET_DVR_PTZCFG_INPUTSTREAM_GET、NET_DVR_PTZCFG_INPUTSTREAM_SET。
- NET_DVR_USING_SERIALPORT(当前使用串口)使用 1 个保留字节新增参数：byProtocolType(串口协议类型)。
- NET_DVR_VCS_USER_INFO(大屏用户信息)使用 1 个保留字节新增参数：byWallNo(开启屏幕管理区域的墙号)。
- NET_DVR_SCREENALARMCFG(多屏控制器报警信息)扩展：
 - 1) byAlarmType(报警类型)新增取值：13-解码板断网恢复、14-解码板 IP 地址冲突恢复。
 - 2) bySubException(输入异常时子异常)新增取值：8-分辨率改变导致输入源不支持 9/16 分屏画面，关闭对应子窗口；9-输出对应的输入源个数超限。
- [NET_DVR_CAM_MODE](#)(输入源枚举类型)新增取值：NET_DVR_CAM_DP、NET_DVR_CAM_HDTVI、NET_DVR_CAM_JOINT。
- NET_DVR_SUBBOARD_INFO(子板信息)中 byBoardType 新增取值：10-DP 输入、11-HDTVI 输入、12-HDBaseT 输出、13-HDMI 输入。
- [NET_DVR_DISPLAYPARAM](#)(显示输出参数)中 byDispChanType(输出口类型)：12-HDBaseT。
- [NET_DVR_DEV_CHAN_INFO_EX](#)(前端编码设备信息)中 byChanType(通道类型)新增取值：5-拼接通道。
- [NET_DVR_SCREEN_CONTROL_V41](#)(屏幕控制参数)中 byProtocol(串口协议类型)新增取值：6-LCD-D20、7-LCD-L2、8-LCD-Z1、9-LCD-D20/D、10-LCD-D20/S、11-LCD-D20/P、12-LCD-D20/T、13-LCD-D20/F、14-LCD-D20/DF、15-LCD-D20/FS、16-LCD-D20/FP、17-LCD-D20/FT、18-LCD-D5022、19-LCD-D5032、20-LCD-D5042+、21-LCD-D20/Z。
- 电视墙能力集([WallAbility](#)，对应接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：WALL_ABILITY)扩展，新增节点：<JointSignal>(信号源拼接能力)、<SupportGetPlayingPlan>(是否支持获取正在执行的预案)、<InputStreamV40>(是否支持本地源 V40 扩展配置)、<InputStreamCut>(是否支持信号源裁剪)。
- RS232 和 RS485 串口能力集(DEVICE_SERIAL_ABILITY)扩展，<RS485>(RS485 串口能力)新增子节点：<SupportConfigWorkMode>(支持配置工作模式)。
- 新增错误码：729~733、900、943~950。

Version 5.1.3.5 (build20150701)

- DS-6400HD-T V3.5.0
- 新增窗口漫游模式开关配置功能（对应接口：[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)）：
命令：NET_DVR_GET_WIN_ROAM_SWITCH_CFG、NET_DVR_SET_WIN_ROAM_SWITCH_CFG。
- [NET_DVR_MATRIX_DECCHAN_CONTROL](#)(窗口参数配置)使用 4 个保留字节新增参数：dwAllCtrlType(所有子窗口一起操作的类型)。
- 设备能力集扩展（对应接口：[NET_DVR_GetDeviceAbility](#)），解码器能力集(DECODER_ABILITY)中新增节点：<SupportWinRoamSwitch>(是否支持窗口漫游)。

Version 5.1.1.6 (build20150409)

- DS-B20 V1.3
- 新增 LOGO 图片下载接口：[NET_DVR_DownloadLogo](#)。
- 新增多相机拼接上墙功能（对应接口：[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)）：
命令：NET_DVR_GET_FUSION_SCALE、NET_DVR_SET_FUSION_SCALE(融合规模配置)；
NET_DVR_GET_FUSION_CFG、NET_DVR_SET_FUSION_CFG(融合参数配置)。

- 新增 Camera 通道绑定串口配置（对应接口：[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)）：
命令：NET_DVR_GET_CAMERACHAN_SERIALCFG、NET_DVR_SET_CAMERACHAN_SERIALCFG。
- 新增 3 维带速度的云台控制功能（对应接口：[NET_DVR_RemoteControl](#)）：
命令：NET_DVR_PTZ_3D_SPEED。
- 新增获取视频综合平台状态功能（对应接口：[NET_DVR_GetDVRConfig](#)）：
命令：NET_DVR_GET_MATRIX_STATUS。
- 新增相机拼接能力集（对应接口：[NET_DVR_GetDeviceAbility](#)）：
能力集类型：CAM_FUSION_ABILITY。
- [NET_DVR_MatrixPicAdjust](#) 图像微调命令 dwPicAdjust 新增取值：DECODEPIC_HEIGHTADJUST(图像高度调整)、DECODEPIC_WIDTHADJUST(图像宽度调整)。
- [NET_DVR_MATRIX_DECCHAN_CONTROL](#)(窗口参数配置)使用 1 个保留字节新增参数：byEnableVcaDec(是否启用智能解码)。
- [NET_DVR_GBT28181_ACCESS_CFG](#)(GBT28181 协议接入配置)使用 1 个保留字节新增参数 szDeviceDomain(设备域)。
- [NET_DVR_MAINBOARD_SERIAL_CFG](#)(主控板串口参数)扩展，使用 3 个保留字节新增参数：wProtocol(串口协议)、byVariable(串口是否可变)，参数 byFunType(串口模式)新增取值：4-PTZ 控制、5-矩阵控制、6-控制台。
- [NET_DVR_SUBSYSTEMINFO_V40](#)(子系统信息)中 bySubSystemType(子系统类型)新增取值：12-超高清输入子系统、13-X86 中心服务板、14-X86 业务服务板、15-网络光纤交换子系统、16-多相机拼接子系统。
- [NET_DVR_SCENE_CONTROL_INFO](#)(场景控制信息)中 dwCmd(控制命令)新增取值：5-删除场景。
- 电视墙能力集([WallAbility](#)，对应接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：WALL_ABILITY)扩展，新增节点：<VoutMotionFluency>(是否支持畅显)、<VcaDecode>(是否支持智能解码)、<DownloadLogo>(是否支持 LOGO 下载)。
- 设备软硬件能力集(DEVICE_SOFTHARDWARE_ABILITY)新增节点：<SupportMainBoardCameraPTZ>(支持主控控制 PTZ)、<Support3DPTZ>(是否支持 3D 带速云台控制)、<SupportUnifiedIPCProtocol>(是否支持统一 IPC 协议)。
- RS232 和 RS485 串口能力集(DEVICE_SERIAL_ABILITY)扩展，新增节点：<SerialProtocol>(可变串口能力)，<RS232>中<workMode>新增取值：alarmBoxMode, matrixControl, screenControl, directConnect, keyboardControl, monitorControl, PTZControl。
- 设备通用能力集(DEVICE_ABILITY_INFO)，其中 GB/T28181 能力(GBT28181AccessAbility)，新增节点：<SupportDeviceDomain>(设备域)。
- 图像参数能力集(DEVICE_VIDEOPIC_ABILITY)中<PicAdjust>(图像微调能力)新增子节点：<heightAdjustParam>(高度调整)、<widthAdjustParam>(宽度调整)。
- 新增设备类型：DS_6441VS_B20_DEV_CLASS(相机拼接类型)。

Version 5.0.3.16 (build20150205)

- DS-C10N V2.1
- [NET_DVR_PLAN_CFG](#)(预案配置)使用 2 个保留字节新增参数：byWallNo(电视墙号)、byPlanNo(预案号)。
- [NET_DVR_VIRTUALLED_PARAM](#)(虚拟 LED 参数)中结构体 NET_DVR_YUVPARAM 修改为 NET_DVR_RGB_COLOR。
- [NET_DVR_VCS_USER_INFO](#)(输入源参数)中 dwRight(远程操作权限)新增取值：bit16-信号源权限。
- 分布式设备类型([NET_DVCS_DEVICE_TYPE](#))新增枚举类型：NET_DVR_SDI_PROC(SDI 处理器)。

Version 5.0.2.15 (build20141212)

- DS-C10S V1.3
- 新增获取子板信息功能（对应接口：[NET_DVR_GetDVRConfig](#)）：
命令：NET_DVR_GET_SUBBOARD_INFO、NET_DVR_GET_SUBBOARD_EXCEPTION。
- [NET_DVR_INPUTSTREAMCFG](#)(输入源参数)使用 1 个保留字节新增参数：byStatus(信号源状态)。
- [NET_DVR_WALLOUTPUTPARAM](#)(电视墙显示输出参数)使用 4 个保留字节新增参数：wLEDWidth(LED 屏输出分辨率宽)、wLEDHeight(LED 屏输出分辨率高)。
- [NET_DVR_VCS_USER_INFO](#)(输入源参数)使用 20 个保留字节新增参数：struManageRegion(屏幕管理区域)。
- 电视墙能力集([WallAbility](#)，对应接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：WALL_ABILITY)扩展，新增节点：<WindowLimit>(窗口限制)，<VoutResource>(支持的分辨率)中新增子节点：<LEDResolution>(LED 输出分辨率)。
- 设备软硬件能力集(DEVICE_SOFTHARDWARE_ABILITY)新增节点：<DeviceSubBoardInfo>(设备子板信息)。
- 设备用户管理能力集(DEVICE_USER_ABILITY)，<VideoWallPermission>(电视墙权限)中<permissionType>新增权限类型：manageScreenArea。
- 新增错误号：928~942。

Version 4.3.1.6 (build201411106)

- DS-B20 V1.2
- 新增获取 IPC 协议列表扩展接口：[NET_DVR_GetIPCProtoList_V41](#)。
- 新增获取窗口关联资源信息功能（对应接口：[NET_DVR_GetDeviceConfig](#)）：
命令：NET_DVR_GET_WINASSOCIATEDDEVINFO。
- 新增音频切换功能（对应接口：[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDeviceConfig](#)）：
命令：NET_DVR_GET_VW_AUDIO_CFG、NET_DVR_SET_VW_AUDIO_CFG。
- 新增场景配置文件导入导出功能（对应接口：[NET_DVR_UploadFile](#)、[NET_DVR_StartDownload](#)）：
命令：SCENE_CONFIGURATION_FILE、NET_SDK_DOWNLOAD_SCENE_CONFIGURATION_FILE。
- 新增主控板串口参数配置（对应接口：[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)）：
NET_DVR_GET_MAINBOARD_SERIAL、NET_DVR_SET_MAINBOARD_SERIAL。
- 新增解码通道 28181 协议编码 ID 配置功能（对应接口：[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDeviceConfig](#)）：
NET_DVR_GET_GBT28181_DECCHANINFO_CFG、NET_DVR_SET_GBT28181_DECCHANINFO_CFG。
- 新增视支持频通道 28181 协议编码 ID 配置功能（对应接口：[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDeviceConfig](#)）：
NET_DVR_GET_GBT28181_CHANINFO_CFG、NET_DVR_SET_GBT28181_CHANINFO_CFG。
- 新增支持 28181 接入配置（对应接口：[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)）：
NET_DVR_GET_GBT28181_ACCESS_CFG、NET_DVR_SET_GBT28181_ACCESS_CFG。
- [NET_DVR_MatrixPicAdjust](#) 图像微调命令 dwPicAdjust 新增取值：DECODEPIC_AUTOADJUST(图像自动调整)。
- [NET_DVR_DISPLAYPARAM](#)(显示输出参数)中 byDispChanType(输出口类型)和
[NET_DVR_WALLOUTPUTPARAM](#)(电视墙显示输出参数)中 byDisplayMode(输出连接模式)分别新增取值：
5- SDI、6- FIBER、7- RGB、8- YPrPb、9- VGA/HDMI/DVI 自适应、10- 3GSDI、11- VGA/DVI 自适应。
- [NET_DVR_SUBSYSTEMINFO_V40](#)(视频综合平台子系统信息)，使用 3 个字节新增参数：
byAudioChanNums(音频通道个数)、byAudioChanStartNum(音频通道起始号)、byAudioChanType(音频通道类型)，byInterfaceType(接口类型)新增取值：7- RGB、8- YPrPb、9- VGA/HDMI/DVI 自适应、10- 3GSDI、11- VGA/DVI 自适应。
- [NET_DVR_WALL_CFG](#)(电视墙全局参数)使用 1 个字节新增参数：byWinStaticMode(窗口解码停止显示模式)。

- [NET_DVR_MATRIX_DECCHAN_CONTROL](#)(窗口参数配置) 使用 3 个字节新增参数: byLowLight(低照度)、byNoiseReduction(3D 降噪)、byDefog(透雾)。
- [NET_DVR_DEV_CHAN_INFO_EX](#)(前端编码设备信息)中 byChanType(通道类型)新增取值: 4- 虚拟屏服务器通道。
- 电视墙能力集([WallAbility](#), 对应接口: [NET_DVR_GetDeviceAbility](#), 能力集类型: WALL_ABILITY)扩展, 新增节点: <windowStaticMode>、<LowLight>、<NoiseReduction>、<Defog>、<AllWinDecSwitch>。
- RS232 和 RS485 串口能力集(DEVICE_SERIAL_ABILITY)新增节点: <VariableSerial>(可变串口能力)。
- 设备图像参数能力集(DEVICE_VIDEOPIC_ABILITY)新增节点: <PicAdjust>(图像调整能力)。

Version 4.3.0.6 (build20140722)

- DS-6400HD-S V2.0.0
- [NET_DVR_SHOW_CONTROL_INFO](#)(电视墙显示编号控制)使用 4 个字节新增参数: dwWallNo(电视墙号)。
- [NET_DVR_VIRTUALLED_PARAM](#)(虚拟 LED 参数) 使用 2 个字节新增参数: byMoveMode(字符移动模式)、byMoveDirection(移动方向)。
- [NET_DVR_DEVICEINFO_V30](#)(设备参数)中 bySupport4 新增取值: bySupport4 & 0x4 表示是否支持拼控统一接口。
- 电视墙能力集([WallAbility](#), 对应接口: [NET_DVR_GetDeviceAbility](#), 能力集类型: WALL_ABILITY)扩展, 新增节点: <VideoEffect>, <VirtualLed>中新增子节点<moveMode>、<moveDirection>。

Version 4.2.8.7 (build20140408)

- DS-C10S V1.2
- 新增电视墙拼接模式配置命令 (对应接口: [NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)):
NET_DVR_GET_VIDEOWALLDISPLAYMODE、NET_DVR_SET_VIDEOWALLDISPLAYMODE
- 新增电视墙显示输出位置配置命令 (对应接口: [NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDeviceConfig](#)):
NET_DVR_GET_VIDEOWALLDISPLAYPOSITION、NET_DVR_SET_VIDEOWALLDISPLAYPOSITION
- 新增获取设备显示输出号命令 (对应接口: [NET_DVR_GetDVRConfig](#)):
NET_DVR_GET_VIDEOWALLDISPLAYNO
- 新增电视墙上编号显示控制命令 (对应接口: [NET_DVR_RemoteControl](#)):
NET_DVR_DISPLAY_CHANNO_CONTROL
- 新增电视墙窗口配置命令 (对应接口: [NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDeviceConfigEx](#)):
NET_DVR_GET_VIDEOWALLWINDOWPOSITION、NET_DVR_SET_VIDEOWALLWINDOWPOSITION
- 新增电视墙关闭所有窗口命令 (对应接口: [NET_DVR_RemoteControl](#)):
NET_DVR_VIDEOWALLWINDOW_CLOSEALL
- 新增获取窗口解码信息的功能 (对应接口: [NET_DVR_GetDeviceConfig](#)):
NET_DVR_GET_WIN_DEC_INFO
- 新增场景模式配置命令 (对应接口: [NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDeviceConfig](#)):
NET_DVR_GET_VW_SCENE_PARAM、NET_DVR_SET_VW_SCENE_PARAM
- 新增获取当前场景号命令 (对应接口: [NET_DVR_GetDeviceConfig](#)):
NET_DVR_GET_CURRENT_SCENE
- 新增场景控制命令 (对应接口: [NET_DVR_RemoteControl](#)):
NET_DVR_SCENE_CONTROL
- 新增虚拟 LED 配置命令 (对应接口: [NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)):
NET_DVR_GET_VIRTUALLED、NET_DVR_SET_VIRTUALLED
- [NET_DVR_CAM_MODE](#)(信号输入源类型)新增枚举类型: NET_DVR_CAM_YPBPR、NET_DVR_CAM_USB、

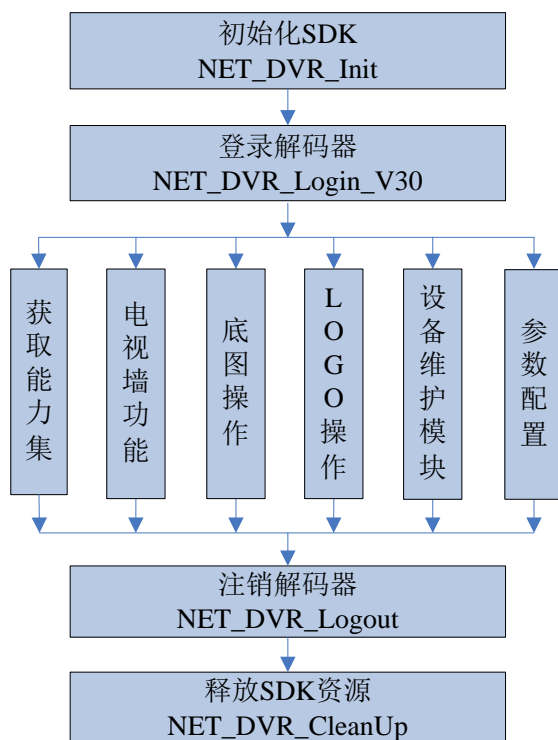
NET_DVR_CAM_SDI、NET_DVR_CAM_HDI。

- [NET_DVR_INPUTSTREAMCFG](#)(输入源参数)中分辨率 wResolutionX、wResolutionY 变更为 dwResolution , byNetSignalResolution 变更为 byRes1。
- [NET_DVR_WALLOUTPUTPARAM](#)(电视墙显示输出参数)使用一个保留字节新增参数: byBackgroundColor(背景色)。
- [NET_DVR_DEV_CHAN_INFO_EX](#)(前端编码设备信息)使用一个保留字节新增参数: dwChannel (通道号), 参数 byChanType 新增类型取值: 3-本地输入源, byDeviceType 新增类型取值: 3-X86 高清输入。
- [NET_DVR_SCREEN_CONTROL_V41](#)(屏幕控制参数)使用 20 个保留字节新增参数: struRect(坐标范围)。
- 电视墙能力集([WallAbility](#), 对应接口: [NET_DVR_GetDeviceAbility](#), 能力集类型: WALL_ABILITY)扩展, 新增节点: <PicViewParam>、<ShowDispChanNo>, <VoutResource >中新增子节点<AllOutputConfigOnly>、<backgroundColor>。

3 函数调用顺序

3.1 解码器主要流程

图 3.1 解码器主要流程



解码器功能模块包括设备能力集、电视墙功能、报警模块、远程设备维护、以及基本参数配置等。

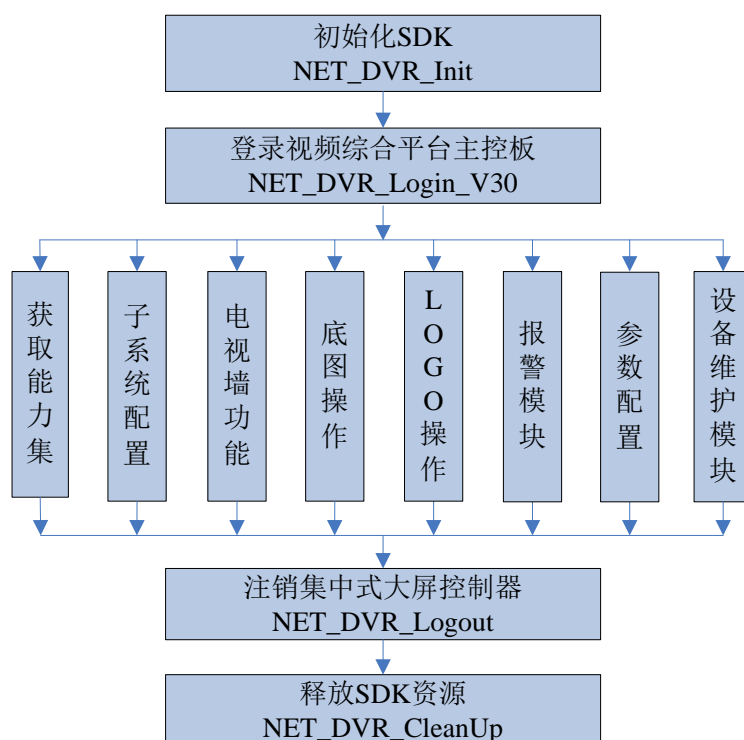
DS-6400HD-T 需要启用窗口漫游开关 ([NET_DVR_WIN_ROAM_SWITCH_CFG](#)) 才支持本文档介绍的电视墙拼控相关接口，如果漫游开关为关闭状态，遵循普通解码器接口，以显示通道和解码通道方式操作设备，相关接口请参考《设备网络 SDK 编程指南（解码器）.pdf》。

- 在初始化 SDK 和登录多屏控制器之后可通过接口 [NET_DVR_GetDeviceAbility](#) (能力集类型: WALL_ABILITY) 获取设备支持的能力集，包括：电视墙拼接屏、窗口、场景、图层个数、基准坐标，以及虚拟 LED、底图、LOGO 功能等。
- 电视墙功能包括电视墙拼接、开窗、解码相关功能，具体实现请参考【[3.4 电视墙功能接口流程](#)】。
- 底图功能：
 - 上传底图图片到设备端：[NET_DVR_PicUpload](#)、[NET_DVR_GetPicUploadProgress](#)、[NET_DVR_GetPicUploadState](#)、[NET_DVR_CloseUploadHandle](#)。上传图片要求 jpg 格式，宽高必须 16 对齐，支持分辨率：1280*720 ~ 7680 * 4320。
 - 底图控制：使用 [NET_DVR_SET_BASEMAP_WIN_CFG](#) (对应接口：[NET_DVR_SetDVRConfig](#)) 先设置底图窗口 ([NET_DVR_BASEMAP_WINCFG](#))，然后通过 [NET_DVR_SET_BASEMAP_CFG](#) (对应接口：[NET_DVR_SetDVRConfig](#)) 控制底图在指定的窗口上显示或隐藏 ([NET_DVR_BASEMAP_CONTROL_CFG](#))。
 - 底图删除：[NET_DVR_RemoteControl](#) (命令：NET_DVR_DELETE_PICTURE)。
- LOGO 功能：

- 上传 LOGO 图片到设备端：[NET_DVR_UploadLogo_NEW](#)。上传的 Logo 图片要求 bmp 格式，24 位图，支持分辨率：32*32 ~ 256*128。
- LOGO 控制：通过 [NET_DVR_SET_WIN_LOGO_CFG](#)（对应接口：[NET_DVR_SetDVRConfig](#)）实现 LOGO 显示控制，包括显示区域、闪烁透明模式等（[NET_DVR_WIN_LOGO_CFG](#)）。通过 [NET_DVR_SET_MATRIX_LOGO_CFG](#)（对应接口：[NET_DVR_SetDVRConfig](#)）可以设置 LOGO 名称（[NET_DVR_MATRIX_LOGO_CFG](#)）。
- LOGO 删除：[NET_DVR_RemoteControl](#)（命令：NET_DVR_DELETE_LOGO）。
- 设备维护模块包括远程升级设备、日志查找、配置文件导入导出、恢复设备默认参数、远程重启相关功能，相关接口参见【[6.28 设备维护](#)】。
- 参数配置：设备参数、时间参数、网络参数、用户参数等其他参数配置，相关接口：[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)。

3.2 视频综合平台主要流程

图 3.2 视频综合平台主要流程



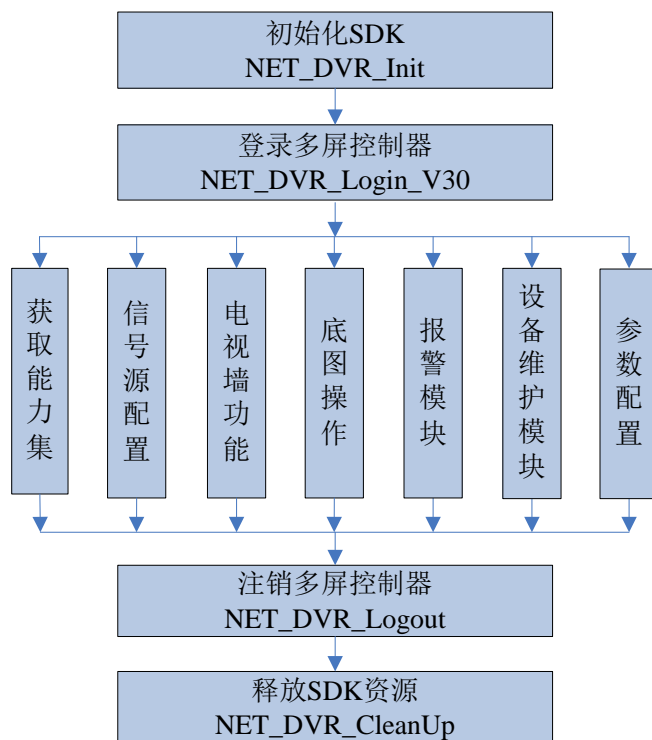
集中式多屏控制器的功能模块包括设备能力集、信号源配置、电视墙功能、报警模块、远程设备维护、以及基本参数配置等。

- 在初始化 SDK 和登录视频综合平台主控板之后可通过接口 [NET_DVR_GetDeviceAbility](#)（能力集类型：WALL_ABILITY）获取设备支持的能力集，包括：输入输出源、电视墙拼接屏、窗口、场景、图层个数、基准坐标，以及虚拟 LED、底图功能等。
- 子系统配置，获取和设置视频综合平台子系统信息，包括编解码子系统网络参数、输入输出通道信息等，相关接口：[NET_DVR_MatrixGetSubSystemInfo_V40](#)、[NET_DVR_MatrixSetSubSystemInfo_V40](#)。
- 电视墙功能包括电视墙拼接、开窗、解码相关功能，具体实现请参考【[3.4 电视墙功能接口流程](#)】。

- 底图功能：
 - 上传底图图片到设备端：[NET_DVR_PicUpload](#)、[NET_DVR_GetPicUploadProgress](#)、[NET_DVR_GetPicUploadState](#)、[NET_DVR_CloseUploadHandle](#)。多屏控制器上传底图后不支持删除命令，重新上传覆盖之前的底图即可。上传图片要求 jpg 格式，宽高必须 16 对齐，支持分辨率：1280*720 ~ 7680 * 4320。
 - 底图控制：使用 NET_DVR_SET_BASEMAP_WIN_CFG（对应接口：[NET_DVR_SetDVRConfig](#)）先设置底图窗口（[NET_DVR_BASEMAP_WINCFG](#)），然后通过 NET_DVR_SET_BASEMAP_CFG（对应接口：[NET_DVR_SetDVRConfig](#)）控制底图在指定的窗口上显示或隐藏（[NET_DVR_BASEMAP_CONTROL_CFG](#)）。
- LOGO 功能：
 - 上传 LOGO 图片到设备端：[NET_DVR_UploadLogo_NEW](#)。上传的 Logo 图片要求 bmp 格式，24 位图，支持分辨率：32*32 ~ 256*128。
 - LOGO 控制：通过 NET_DVR_SET_WIN_LOGO_CFG（对应接口：[NET_DVR_SetDVRConfig](#)）实现 LOGO 显示控制，包括显示区域、闪烁透明模式等（[NET_DVR_WIN_LOGO_CFG](#)）。通过 NET_DVR_SET_MATRIX_LOGO_CFG（对应接口：[NET_DVR_SetDVRConfig](#)）可以设置 LOGO 名称（[NET_DVR_MATRIX_LOGO_CFG](#)）。
 - LOGO 删除：[NET_DVR_RemoteControl](#)（命令：NET_DVR_DELETE_LOGO）。
- 报警模块包括报警布防和报警监听两种模式，用于接收设备上传的报警信息，具体流程请参考【[3.5 报警模块流程](#)】。
- 参数配置：设备参数、时间参数、网络参数、用户参数等其他参数配置，相关接口：[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)。
- 设备维护模块包括远程升级设备、日志查找、配置文件导入导出、恢复设备默认参数、远程重启相关功能，相关接口参见【[6.28 设备维护](#)】。

3.3 多屏控制器主要流程

图 3.3 多屏控制器主要流程



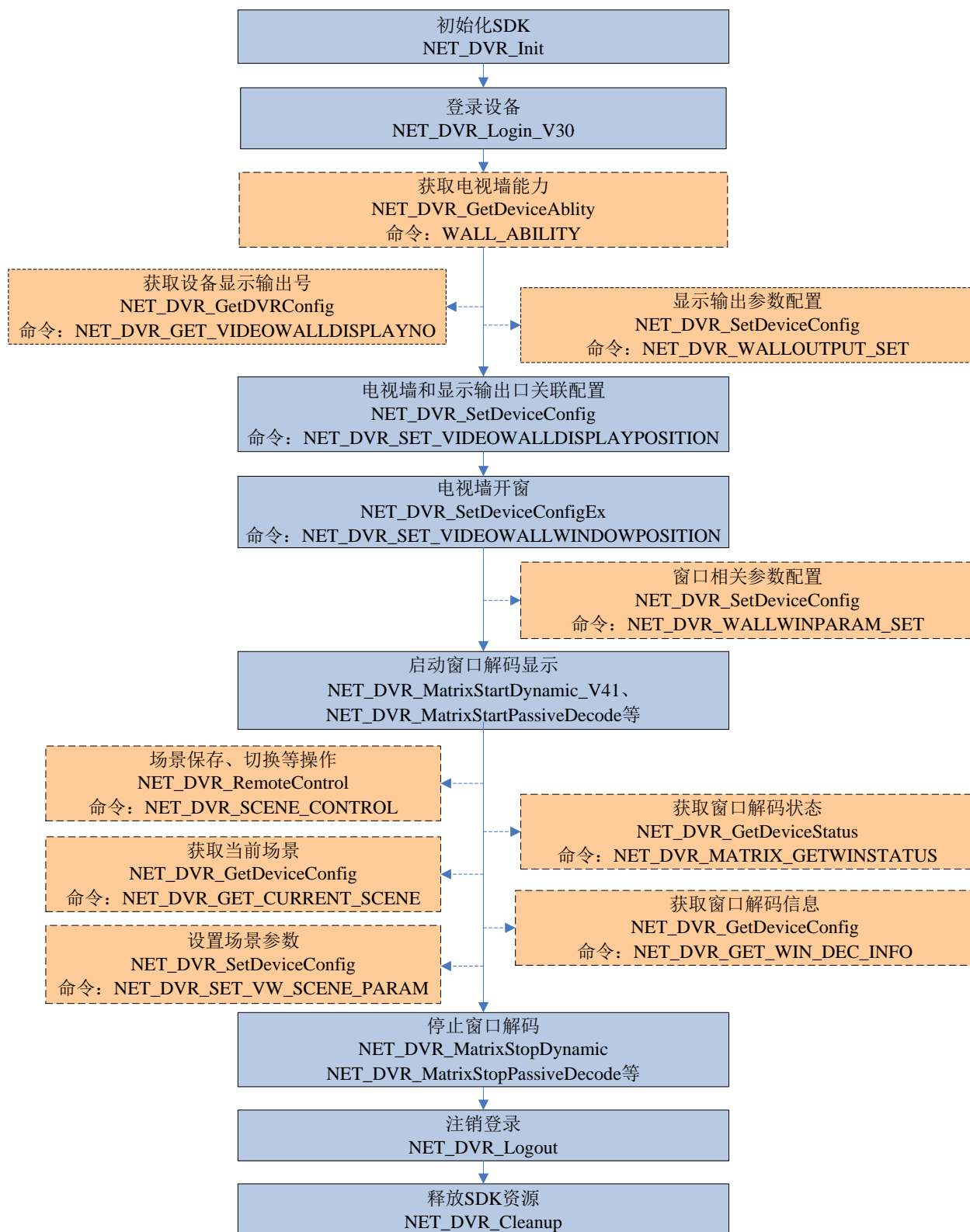
多屏控制器的功能模块包括设备能力集、信号源（包括子板、子设备信息）配置、电视墙功能、报警模块、远程设备维护、以及基本参数配置等。

- 在初始化 SDK 和登录多屏控制器之后可通过接口 [NET_DVR_GetDeviceAbility](#)（能力集类型：WALL_ABILITY）获取设备支持的能力集，包括：输入输出源、电视墙拼接屏、窗口、场景、图层个数、基准坐标，以及虚拟 LED、底图功能等。
- 信号源配置
 - 集中式多屏控制器子板信息通过 [NET_DVR_GetDVRConfig](#)（命令：NET_DVR_GET_SUBBOARD_INFO）获取，包括子板类型、槽位号、接口个数等。分布式多屏控制器子设备信息通过 [NET_DVR_GetDevList](#) 获取，包括设备版本、类型、状态、网络地址、序号等。
 - 通过命令 NET_DVR_GET_DEV_NETINFO、NET_DVR_SET_DEV_NETINFO 可配置控制器中集中式解码板或分布式子设备的网络参数，对应接口：[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDeviceConfig](#)。
 - 网络信号源、本地信号源、矩阵信号源等输入信号源列表可以通过接口 [NET_DVR_GetInputSignalList_V40](#) 获取。
网络信号源：使用动态解码接口进行实时操作，或者通过客户端（WEB 或者 iPad）实现网络信号源的添加、删除或者修改等管理。
本地信号源：通过命令 NET_DVR_SET_INPUTSTREAMCFG_V40（对应接口：[NET_DVR_SetDVRConfig](#)）进行设置。
矩阵信号源：配置命令为 NET_DVR_GET_EXTERNAL_MATRIX_CFG、NET_DVR_SET_EXTERNAL_MATRIX_CFG，分别对应接口：[NET_DVR_GetDeviceConfig](#)、[NET_DVR_SetDeviceConfig](#)。
- 电视墙功能包括电视墙拼接、开窗、解码相关功能，具体实现请参考【[3.4 电视墙功能接口流程](#)】。

- 底图功能：
 - 上传底图图片到设备端：[NET_DVR_PicUpload](#)、[NET_DVR_GetPicUploadProgress](#)、[NET_DVR_GetPicUploadState](#)、[NET_DVR_CloseUploadHandle](#)。多屏控制器上传底图后不支持删除命令，重新上传覆盖之前的底图即可。
 - 底图控制：使用 `NET_DVR_SET_BASEMAP_WIN_CFG`（对应接口：[NET_DVR_SetDVRConfig](#)）先设置底图窗口（[NET_DVR_BASEMAP_WINCFG](#)），然后通过 `NET_DVR_SET_BASEMAP_CFG`（对应接口：[NET_DVR_SetDVRConfig](#)）控制底图在指定的窗口上显示或隐藏（[NET_DVR_BASEMAP_CONTROL_CFG](#)）。
- 报警模块包括报警布防和报警监听两种模式，用于接收设备上传的报警信息。多屏控制器仅支持报警布防模式，具体流程请参考【[3.5.1 报警布防](#)】。
- 设备维护模块包括远程升级设备、日志查找、配置文件导入导出、恢复设备默认参数、远程重启相关功能，相关接口参见【[6.28 设备维护](#)】。
- 参数配置：设备参数、时间参数、网络参数、用户参数等其他参数配置，相关接口：[NET_DVR_GetDVRConfig](#)、[NET_DVR_SetDVRConfig](#)。

3.4 电视墙功能接口流程

图 3.4 解码上墙调用流程



以下主要介绍电视墙功能配置，其中提及的 SET 命令一般都有对应的 GET 命令，建议调用 [NET_DVR_SetDVRConfig](#)、[NET_DVR_SetDeviceConfig](#) 设置某参数之前都先调用 [NET_DVR_GetDVRConfig](#)、[NET_DVR_GetDeviceConfig](#) 使用相应的 GET 命令获取完整的参数结构，然后修改需更改的参数，作为设置时的输入参数。

- 登录设备、获取设备电视墙能力集，包括基准坐标、支持的屏幕/窗口/画面分割个数等信息。然后获取通过 [NET_DVR_GetDVRConfig](#)（命令：[NET_DVR_GET_VIDEOWALLDISPLAYNO](#)）可获取设备中各个显示屏对应的显示输出口号和接口类型（[NET_DVR_DISPLAYCFG](#)），选择需要的屏幕进行拼接。通过 [NET_DVR_SetDeviceConfig](#)（命令：[NET_DVR_WALLOUTPUT_SET](#)）可以设置各显示屏的输出分辨率、背景色等参数（[NET_DVR_WALLOUTPUTPARAM](#)）。
- 电视墙操作：
 - 通过 [NET_DVR_SetDVRConfig](#)（命令：[NET_DVR_SET_VIDEOWALLDISPLAYMODE](#)）设置拼接模式（[NET_DVR_VIDEOWALLDISPLAYMODE](#)），包括电视墙号、名称、拼接区域等。解码器、视频综合平台和分布式多屏控制器不支持该功能，该步骤省略。
 - 通过 [NET_DVR_SetDeviceConfig](#)（命令：[NET_DVR_SET_VIDEOWALLDISPLAYPOSITION](#)）设置显示输出位置（[NET_DVR_VIDEOWALLDISPLAYPOSITION](#)），即关联电视墙和显示输出口，选择拼接的屏幕以及设置屏幕在电视墙上的位置。
 - 电视墙拼接之后可以通过 [NET_DVR_SetDVRConfig](#)（命令：[NET_DVR_SET_VIRTUALLED](#)）设置虚拟 LED（[NET_DVR_VIRTUALLED_PARAM](#)）。
- 电视墙窗口操作：
 - 通过 [NET_DVR_SetDeviceConfigEx](#)（命令：[NET_DVR_SET_VIDEOWALLWINDOWPOSITION](#)）设置电视墙窗口参数，实现在拼接后的电视墙屏幕上开窗。
 - 开窗后，使用相同的接口和命令修改相关参数即可实现对已开窗口实现漫游窗控制（解码过程中也支持设置）：修改窗口参数（[NET_DVR_VIDEOWALLWINDOWPOSITION](#)）的使能参数（byEnable）即进行开窗或者关闭窗口的操作；修改窗口参数（[NET_DVR_VIDEOWALLWINDOWPOSITION](#)）的目的窗口（struRect）中坐标或宽高值即进行移动或缩放窗口的操作。通过接口 [NET_DVR_RemoteControl](#) 可以对窗口实现置顶置底的操作或者关闭电视墙上所有窗口。
 - 开窗后，需要进行窗口相关参数配置：[NET_DVR_SetDVRConfig](#)（命令：[NET_DVR_WALLWINPARAM_SET](#)）设置电视墙窗口参数（[NET_DVR_WALLWINPARAM](#)），比如分屏模式（1、4、9、16 等画面分割设置，能力集中获取）等。配置分屏模式之后，每一个子窗口对应一个子窗口号，从 1 开始，从左往右、从上往下递增。
- 完成参数配置后即可启动窗口上墙显示：
 - 动态解码，相关接口：[NET_DVR_MatrixStartDynamic_V41](#)、[NET_DVR_MatrixStopDynamic](#)，切换解码前端源时不需要停止解码，修改动态解码参数（[NET_DVR_PU_STREAM_CFG_V41](#)）再次调用动态解码接口即可（多次调用动态解码接口以最后一次的为准）。网络信号源、本地信号源选择均在动态解码接口的参数 [NET_DVR_PU_STREAM_CFG_V41](#) 中设置。
 - 被动解码，相关接口：[NET_DVR_MatrixStartPassiveDecode](#)、[NET_DVR_MatrixSendData](#)、[NET_DVR_MatrixStopPassiveDecode](#)。集中式多屏控制器被动解码只支持文件流，分布式多屏控制器不支持被动解码。
 - 解码上墙可以通过 [NET_DVR_MatrixSetDecChanEnable](#) 控制开关。
- 场景操作：
 - 通过 [NET_DVR_RemoteControl](#)（命令：[NET_DVR_SCENE_CONTROL](#)）可以进行场景保存、切换、初始化操作。

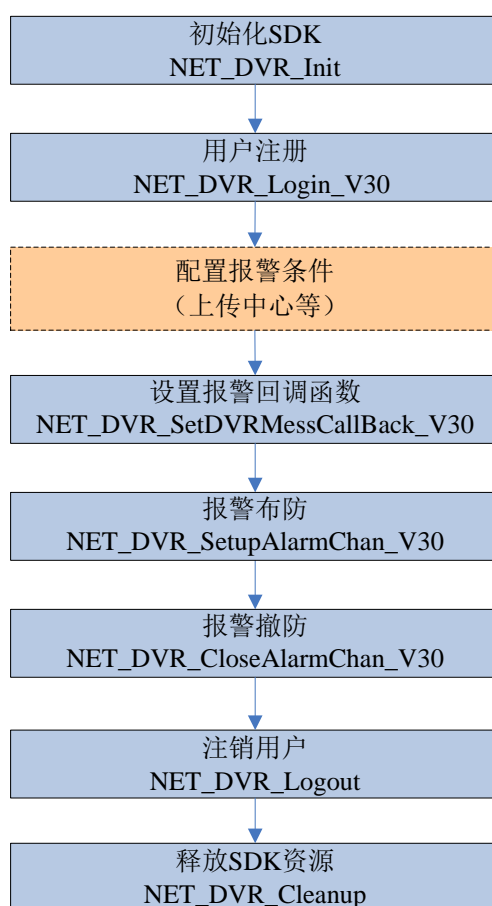
- 保存场景后，可以通过 [NET_DVR_SetDeviceConfig](#)（命令：NET_DVR_SET_VW_SCENE_PARAM）设置场景参数，比如修改场景名称等。
- 通过 [NET_DVR_GetDeviceConfig](#)（命令：NET_DVR_GET_CURRENT_SCENE）可以获取当前正在使用的场景模式。
- 状态获取：通过 [NET_DVR_GetDeviceConfig](#)（命令：NET_DVR_GET_WIN_DEC_INFO）可以获取各窗口当前的解码信息（[NET_DVR_MATRIX_DEC_CHAN_INFO_V41](#)），包括窗口关联的信号源、主动还是被动解码状态等。而解码的码流格式、帧率、分辨率、已解码帧数等窗口状态信息（[NET_DVR_WALL_WIN_STATUS](#)）可以通过 [NET_DVR_GetDeviceStatus](#)（命令：NET_DVR_MATRIX_GETWINSTATUS）获取。

[调用实例代码](#)

3.5 报警模块流程

3.5.1 报警布防

图 3.5 报警布防调用流程

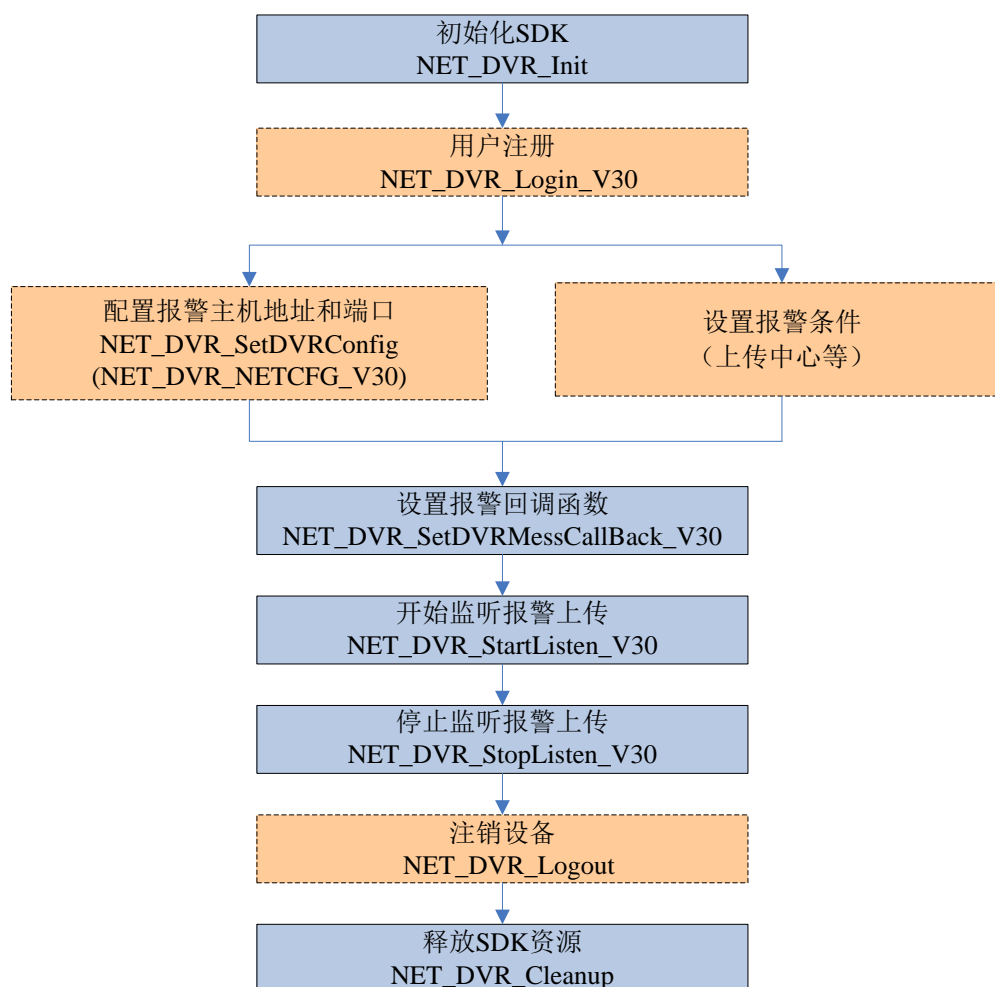


- “布防”报警方式是指 SDK 主动连接设备，并发起报警上传命令，设备发生报警立即发送给 SDK。
- “布防”方式需要先进行用户注册（[NET_DVR_Login_V40](#)），然后需要设置报警回调函数（[NET_DVR_SetDVRMessageCallBack_V30](#)）和对设备进行布防（[NET_DVR_SetupAlarmChan_V41](#)）。报警信息在设置的报警回调函数中获取。整个报警上传过程结束后还需要调用撤防接口等操作，释放相关资源。

- 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，视频综合平台报警（温度超限、热插拔、风扇等异常）参数配置，相关的接口：[NET_DVR_GetDVRConfig](#)（命令：NET_DVR_VIDEOPLATFORMALARMCFG_GET）、[NET_DVR_SetDVRConfig](#)（命令：NET_DVR_VIDEOPLATFORMALARMCFG_SET）。相关参数如果已经配置完成，那么虚线框部分可以省略。

3.5.2 报警监听

图 3.6 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。

3.6 电视墙号、窗口号等编号说明

电视墙功能各接口和结构体大多涉及电视墙号、窗口号、底图窗口号、显示输出号等编号，由此实现多窗口、多显示器灵活切换控制。

本文中提及的各种编号说明如下：

1) 电视墙号、窗口号、底图窗口号、显示输出号、输入口号、底图序号、LOGO 序号、虚拟 LED 号、子板槽位序号、信号源序号、光口序号、环网序号、编码通道序号、输入板槽位序号等为实际的顺序编码，从 1 开始，1、2、3、...依次递增，其取值范围通过电视墙能力集获取。

2) 电视墙号(组合)、窗口号(组合)、子窗口号(组合)等组合，都是 4 个字节，其取值：

- 电视墙号(组合)：1 字节电视墙号+3 字节保留，例如： $1 \ll 24 \mid 0$
- 窗口号(组合)：1 字节电视墙号+1 字节保留+2 字节窗口号，例如： $1 \ll 24 \mid 0 \ll 16 \mid 1$
- 子窗口号(组合)：1 字节电视墙号+1 字节子窗口号+2 字节窗口号，例如： $1 \ll 24 \mid 1 \ll 16 \mid 1$
- 底图窗口号(组合)：1 字节电视墙号+1 字节保留+2 字节底图窗口号，例如： $1 \ll 24 \mid 0 \ll 16 \mid 1$
- 显示输出号(组合)：1 字节设备号+1 字节子板号+2 字节显示输出号（分布式多屏控制器是显示输出子系统号），例如： $1 \ll 24 \mid 1 \ll 16 \mid 1$
- 输入口号(组合)：1 字节设备号+1 字节子板号+2 字节输入口号，例如： $1 \ll 24 \mid 1 \ll 16 \mid 1$
- 虚拟 LED 号(组合)：1 字节墙号+1 字节保留+2 字节虚拟 LED 号，例如： $1 \ll 24 \mid 0 \ll 16 \mid 1$
- 子板槽位号(组合)：1 字节子板类型分类+3 字节子板槽位序号，其中子板类型分类：0- 主控，0x1- 输入类型，0x2- 输出类型，例如： $1 \ll 24 \mid 1$
- 光口号(组合)：1 字节设备号+1 字节子板号+2 字节光口序号，例如： $1 \ll 24 \mid 1 \ll 16 \mid 1$
- 环网号(组合)：1 字节设备号+1 字节子板号+2 字节环网序号，例如： $1 \ll 24 \mid 1 \ll 16 \mid 1$
- 编码通道号(组合)：1 字节设备号+1 字节子板号+2 字节编码通道序号，例如： $1 \ll 24 \mid 1 \ll 16 \mid 1$
- 输入板槽位号(组号)：1 字节设备号+3 字节槽位序号（物理槽位号），例如： $1 \ll 24 \mid 1$

4 函数调用实例

4.1 解码上墙示例代码

[相关模块流程图](#)

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

#define XML_BUF 3*1024*1024

void CALLBACK g_ExceptionCallBack(DWORD dwType, LONG IUserID, LONG IHandle, void *pUser)
{
    char tempbuf[256] = {0};
    switch(dwType)
    {
        case EXCEPTION_RECONNECT:    //预览时重连
            printf("-----reconnect-----%d\n", time(NULL));
            break;
        default:
            break;
    }
}

void main() {
    DWORD dwRet;
    //-----
    // 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);

    //设置异常消息回调函数
    NET_DVR_SetExceptionCallBack_V30(0, NULL, g_ExceptionCallBack, NULL);

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

```

NET_DVR_DEVICEINFO_V30 struDeviceInfo;
IUserID = NET_DVR_Login_V30("172.9.31.205", 8000, "admin", "12345", &struDeviceInfo);
if (IUserID < 0)
{
    printf("NET_DVR_Login_V30 failed, err: %d\n", NET_DVR_GetLastError());
    NET_DVR_Cleanup();
    return;
}

//获取电视墙能力集，包括基准坐标、支持的屏幕/窗口/画面分割个数
char* m_pXMLBuf = new char[XML_BUF];
if (!NET_DVR_GetDeviceAbility(IUserID, WALL_ABILITY, NULL, 0, m_pXMLBuf, XML_BUF))
{
    printf("NET_DVR_GetDeviceAbility failed, err: %d\n", NET_DVR_GetLastError());
}

//获取设备显示输出号
NET_DVR_DISPLAYCFG struDisplayCfg = {0};
struDisplayCfg.dwSize = sizeof(struDisplayCfg);
if (!NET_DVR_GetDVRConfig(IUserID, NET_DVR_GET_VIDEOWALLDISPLAYNO, 0, &struDisplayCfg, sizeof(struDisplayCfg),
&dwRet))
{
    printf("NET_DVR_GET_VIDEOWALLDISPLAYNO failed, err: %d\n", NET_DVR_GetLastError());
}

/*****
/* 该 Demo 为简单控制台 Demo，此处为了简化代码，后面配置直接赋值
/* 假设，通过能力集获取到设备最大支持 72 个屏，基准坐标为 1920*1920
/* Demo 中选择前 4 个屏实现 2*2 拼接，开 2 个窗口，选择其中一个窗口解码 1 路图像
/* 4 个屏关联的显示输出号分别为：0x01000001、0x01000002、0x01000003、0x01000004，均为 HDMI 显示器
*****/

//1 字节墙号+ 1 字节(保留或子窗口)+ 2 字节(窗口号)
DWORD dwWallNo = 1 << 24; //拼接时窗口号和子窗口号均为 0

//-----
//设置电视墙拼接模式（解码器和视频综合平台不支持该功能，该步骤省略）
struDisplayMode.byEnable=1;
struDisplayMode.struRect.dwXCoordinate=0; //起始横坐标，基准坐标的倍数
struDisplayMode.struRect.dwYCoordinate=0; //起始纵坐标，基准坐标的倍数
struDisplayMode.struRect.dwWidth=1920*2; //2*2 拼接模式，所以设置基准坐标的 2 倍
struDisplayMode.struRect.dwHeight=1920*2; //2*2 拼接模式，所以设置基准坐标的 2 倍
strcpy((char *)struDisplayMode.sName,"WallTest01"); //电视墙名称

```



```

    if (!NET_DVR_SetDVRConfig(IUserID, NET_DVR_SET_VIDEOWALLDISPLAYMODE, dwWallNo, &struDisplayMode,
sizeof(struDisplayMode)))
    {
        printf("NET_DVR_SET_VIDEOWALLDISPLAYMODE failed, err: %d\n", NET_DVR_GetLastError());
    }

    //-----
    // 获取电视墙显示输出位置，获取 4 个屏幕信息
    DWORD m_dwDispNum=4;

    DWORD m_IDispChan[4]; //显示输出号，通过 NET_DVR_GET_VIDEOWALLDISPLAYNO 获取
    m_IDispChan[0]= 0x01000001;
    m_IDispChan[1]= 0x01000002;
    m_IDispChan[2]= 0x01000003;
    m_IDispChan[3]= 0x01000004;

    DWORD m_dwStatus[4];
    NET_DVR_VIDEOWALLDISPLAYPOSITION m_struWallDispPos[4];

    memset(m_dwStatus, 0, sizeof(m_dwStatus));
    memset(m_struWallDispPos, 0, sizeof(m_struWallDispPos));

    if (!NET_DVR_GetDeviceConfig(IUserID, NET_DVR_GET_VIDEOWALLDISPLAYPOSITION, m_dwDispNum, m_IDispChan, \
        sizeof(DWORD) * m_dwDispNum, m_dwStatus, m_struWallDispPos, m_dwDispNum*
sizeof(NET_DVR_VIDEOWALLDISPLAYPOSITION)))
    {
        printf("NET_DVR_GET_VIDEOWALLDISPLAYPOSITION failed, err: %d\n", NET_DVR_GetLastError());
    }

    //-----
    //设置电视墙显示输出位置，设置 4 个显示屏幕在电视墙上的位置

    m_struWallDispPos[0].byEnable=1;
    m_struWallDispPos[0].dwVideoWallNo= dwWallNo;
    m_struWallDispPos[0].struRectCfg.dwXCoordinate=0;
    m_struWallDispPos[0].struRectCfg.dwYCoordinate=0;

    m_struWallDispPos[1].byEnable=1;
    m_struWallDispPos[1].dwVideoWallNo= dwWallNo;
    m_struWallDispPos[1].struRectCfg.dwXCoordinate=1920;
    m_struWallDispPos[1].struRectCfg.dwYCoordinate=0;

    m_struWallDispPos[2].byEnable=1;
    m_struWallDispPos[2].dwVideoWallNo= dwWallNo;

```

```

m_struWallDispPos[2].struRectCfg.dwXCoordinate=0;
m_struWallDispPos[2].struRectCfg.dwYCoordinate=1920;

m_struWallDispPos[3].byEnable=1;
m_struWallDispPos[3].dwVideoWallNo= dwWallNo;
m_struWallDispPos[3].struRectCfg.dwXCoordinate=1920;
m_struWallDispPos[3].struRectCfg.dwYCoordinate=1920;

if (!NET_DVR_SetDeviceConfig(IUserID, NET_DVR_SET_VIDEOWALLDISPLAYPOSITION, m_dwDispNum, m_IDispChan, \
    sizeof(DWORD) * m_dwDispNum, m_dwStatus, m_struWallDispPos, m_dwDispNum* \
sizeof(NET_DVR_VIDEOWALLDISPLAYPOSITION)))
{
    printf("NET_DVR_SET_VIDEOWALLDISPLAYPOSITION failed, err: %d\n", NET_DVR_GetLastError());
}

//-----
//获取电视墙窗口参数
DWORD m_dwWinCount; //窗口个数
char *pTemp = new char[4 + 256 * sizeof(NET_DVR_VIDEOWALLWINDOWPOSITION)];
memset(pTemp, 0, 4 + 256 * sizeof(NET_DVR_VIDEOWALLWINDOWPOSITION));

//获取电视墙所有窗口信息，只需传电视墙号
if (!NET_DVR_GetDeviceConfig(IUserID, NET_DVR_GET_VIDEOWALLWINDOWPOSITION, 0xffffffff, &dwWallNo, \
    sizeof(DWORD), NULL, pTemp, 4 + 256 * sizeof(NET_DVR_VIDEOWALLWINDOWPOSITION)))
{
    printf("NET_DVR_GET_VIDEOWALLWINDOWPOSITION error, err: %d\n", NET_DVR_GetLastError());
}
m_dwWinCount = *((DWORD*)pTemp); //返回的前 4 个字节为窗口个数
delete []pTemp;

//-----
//设置电视墙窗口参数，即开窗，坐标在电视墙范围之内

m_dwWinCount=2; //开两个窗口
DWORD m_dwWinNum[2];
DWORD m_dwWinStatus[2];
NET_DVR_VIDEOWALLWINDOWPOSITION m_struWinPos[2];
memset(m_struWinPos, 0, 2*sizeof(NET_DVR_VIDEOWALLWINDOWPOSITION));

//1 字节墙号 + 1 字节保留 + 2 字节窗口号
m_dwWinNum[0] = 1<<24; //开窗时窗口号为 0
m_dwWinNum[1] = 1<<24;

m_struWinPos[0].dwSize=sizeof(NET_DVR_VIDEOWALLWINDOWPOSITION);

```

```

m_struWinPos[0].byEnable=1;
m_struWinPos[0].struRect.dwXCoordinate=100;
m_struWinPos[0].struRect.dwYCoordinate=100;
m_struWinPos[0].struRect.dwWidth=1280;
m_struWinPos[0].struRect.dwHeight=720;

m_struWinPos[1].dwSize=sizeof(NET_DVR_VIDEOWALLWINDOWPOSITION);
m_struWinPos[1].byEnable=1;
m_struWinPos[1].struRect.dwXCoordinate=1920;
m_struWinPos[1].struRect.dwYCoordinate=1920;
m_struWinPos[1].struRect.dwWidth=1600;
m_struWinPos[1].struRect.dwHeight=1200;

NET_DVR_IN_PARAM struInputPapam ={0};
NET_DVR_OUT_PARAM struOutputPapam ={0};
DWORD dwRetWinNum[2]; //开窗时窗口号由设备返回

struInputPapam.struCondBuf.pBuf = m_dwWinNum; //输入条件缓冲区
struInputPapam.struCondBuf.nLen = m_dwWinCount * sizeof(DWORD);
struInputPapam.dwRecvTimeout = 5000; //接收超时时间
struInputPapam.struInParamBuf.pBuf = m_struWinPos;
struInputPapam.struInParamBuf.nLen = m_dwWinCount * sizeof(NET_DVR_VIDEOWALLWINDOWPOSITION);
struOutputPapam.lpStatusList = m_dwWinStatus;
struOutputPapam.struOutBuf.pBuf = dwRetWinNum;
struOutputPapam.struOutBuf.nLen = m_dwWinCount*sizeof(DWORD);

if (!NET_DVR_SetDeviceConfigEx(IUserID, NET_DVR_SET_VIDEOWALLWINDOWPOSITION, m_dwWinCount, &struInputPapam,
&struOutputPapam))
{
    printf("NET_DVR_SET_VIDEOWALLWINDOWPOSITION error, err: %d\n", NET_DVR_GetLastError());
}

//动态解码
NET_DVR_PU_STREAM_CFG_V41 m_struDynamicInfo={0};
m_struDynamicInfo.dwSize=sizeof(m_struDynamicInfo);
m_struDynamicInfo.byStreamMode=1; /*直接从设备取流*/
strcpy((char *)m_struDynamicInfo.uDecStreamMode.struDecStreamDev.struDevChanInfo.byAddress, "172.6.22.92");/*前端设
备 IP 地址*/
m_struDynamicInfo.uDecStreamMode.struDecStreamDev.struDevChanInfo.wDVRPort=8000; /*前端设备服务端口*/
strcpy((char *)m_struDynamicInfo.uDecStreamMode.struDecStreamDev.struDevChanInfo.sUserName,"admin"); /*前端设备登
录用户名*/
strcpy((char *)m_struDynamicInfo.uDecStreamMode.struDecStreamDev.struDevChanInfo.sPassword,"12345"); /*前端设备登
录密码*/
m_struDynamicInfo.uDecStreamMode.struDecStreamDev.struDevChanInfo.byChannel=1; /*取通道 1 的码流*/

```

```
m_struDynamicInfo.uDecStreamMode.struDecStreamDev.struDevChanInfo.byFactoryType=0; /*厂家私有协议*/

DWORD dwChannelEx=0x01010001;
//1 字节墙号+1 字节子窗口号+2 字节窗口号: (byWallNo << 24) | (dwSubWinNo<<16) | dwWinNo;

if (!NET_DVR_MatrixStartDynamic_V41(lUserID, dwChannelEx, &m_struDynamicInfo))
{
    printf("NET_DVR_MatrixStartDynamic_V41 error, err: %d\n", NET_DVR_GetLastError());
}

//注销用户
NET_DVR_Logout(lUserID);
//释放 SDK 资源
NET_DVR_Cleanup();

return;
}
```

5 功能接口介绍

5.1 通用接口介绍

功能	接口	支持的产品
初始化 SDK	NET_DVR_Init	全部
释放 SDK 资源	NET_DVR_Cleanup	全部
设置 SDK 网络连接超时时间和连接尝试次数	NET_DVR_SetConnectTime	全部
设置 SDK 重连功能	NET_DVR_SetReconnect	全部
设置接收超时时间	NET_DVR_SetRecvTimeOut	全部
通过解析服务器，获取设备的动态 IP 地址	NET_DVR_GetDVRIPByResolveSvr_Ex	全部
获取所有 IP，用于支持多网卡接口	NET_DVR_GetLocalIP	全部
选择绑定 IP	NET_DVR_SetValidIP	全部
注册接收异常、重连等消息的窗口句柄或回调函数	NET_DVR_SetExceptionCallBack_V30	全部
获取 SDK 的版本号和 build 信息	NET_DVR_GetSDKBuildVersion	全部
获取当前 SDK 的状态信息	NET_DVR_GetSDKState	全部
获取当前 SDK 的功能信息	NET_DVR_GetSDKAbility	全部
获取 SDK 本地参数	NET_DVR_GetSDKLocalCfg	全部
设置 SDK 本地参数	NET_DVR_SetSDKLocalCfg	全部
启用 SDK 写日志文件	NET_DVR_SetLogToFile	全部
返回最后操作的错误码	NET_DVR_GetLastError	全部
返回最后操作的错误码信息	NET_DVR_GetErrorMsg	全部
注册（登录设备）	NET_DVR_Login_V40	全部
注销登录	NET_DVR_Logout	全部

5.2 多路解码器

多路解码器包括但不限于产品型号：DS-6400HD-S、DS-6400HD-T、DS-6900UD。其中 DS-6400HD-T 需要启用窗口漫游开关才支持本文档介绍的电视墙拼控相关接口，如果漫游开关为关闭状态，遵循普通解码器接口，以显示通道和解码通道方式操作设备，相关接口请参考《设备网络 SDK 编程指南（解码器）.pdf》。

多路解码器除支持通用接口的功能外，还支持以下功能接口：

功能	接口	相关参数
获取设备能力集		
获取设备软硬件能力	NET_DVR_GetDeviceAbility	能力集类型：DEVICE_SOFTWARE_ABILITY，

		能力集：软硬件能力 XML 描述
获取设备网络应用参数能力	NET_DVR_GetDeviceAbility	能力集类型：DEVICE_NETAPP_ABILITY， 能力集：网络应用参数能力 XML 描述
获取设备用户管理参数能力	NET_DVR_GetDeviceAbility	能力集类型：DEVICE_USER_ABILITY， 能力集：用户管理参数能力 XML 描述
获取电视墙能力	NET_DVR_GetDeviceAbility	能力集类型：WALL_ABILITY， 能力集：电视墙能力 XML 描述
通用参数配置		
获取设备参数	NET_DVR_GetDVRConfig	命令：NET_DVR_GET_DEVICECFG_V40 结构体： NET_DVR_DEVICECFG_V40
设置设备参数	NET_DVR_SetDVRConfig	命令：NET_DVR_SET_DEVICECFG_V40 结构体： NET_DVR_DEVICECFG_V40
获取时间参数	NET_DVR_GetDVRConfig	命令：NET_DVR_GET_TIMECFG 结构体： NET_DVR_TIME
设置时间参数	NET_DVR_SetDVRConfig	命令：NET_DVR_SET_TIMECFG 结构体： NET_DVR_TIME
获取时区和夏时制参数	NET_DVR_GetDVRConfig	命令：NET_DVR_GET_ZONEANDDST 结构体： NET_DVR_ZONEANDDST
设置时区和夏时制参数	NET_DVR_SetDVRConfig	命令：NET_DVR_SET_ZONEANDDST 结构体： NET_DVR_ZONEANDDST
获取网络参数	NET_DVR_GetDVRConfig	命令：NET_DVR_GET_NETCFG_V30 结构体： NET_DVR_NETCFG_V30
设置网络参数	NET_DVR_SetDVRConfig	命令：NET_DVR_SET_NETCFG_V30 结构体： NET_DVR_NETCFG_V30
获取异常参数	NET_DVR_GetDVRConfig	命令：NET_DVR_GET_EXCEPTIONCFG_V30 结构体： NET_DVR_EXCEPTION_V30
设置异常参数	NET_DVR_SetDVRConfig	命令：NET_DVR_SET_EXCEPTIONCFG_V30 结构体： NET_DVR_EXCEPTION_V30
获取用户参数	NET_DVR_GetDVRConfig	命令：NET_DVR_GET_USERCFG_V40 结构体： NET_DVR_USER_V40
设置用户参数	NET_DVR_SetDVRConfig	命令：NET_DVR_SET_USERCFG_V40 结构体： NET_DVR_USER_V40
获取解码器窗口漫游开关参数	NET_DVR_GetDVRConfig	命令：NET_DVR_GET_WIN_ROAM_SWITCH_CFG 结构体： NET_DVR_WIN_ROAM_SWITCH_CFG
设置解码器窗口漫游开关参数	NET_DVR_SetDVRConfig	命令：NET_DVR_SET_WIN_ROAM_SWITCH_CFG 结构体： NET_DVR_WIN_ROAM_SWITCH_CFG
获取设备支持的 IPC 协议列表		
获取 IPC 协议列表	NET_DVR_GetIPCProtoList_V41	
屏幕服务器登录参数配置		

屏幕服务器登录参数配置	NET_DVR_STDXMLConfig	透传 XML 协议，分别通过 GET、PUT、POST、DELETE 命令获取和设置、添加、删除参数以及获取能力 (69UD V2.0 新增支持)
电视墙操作		
获取设备显示输出号	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_VIDEOWALLDISPLAYNO 结构体: NET_DVR_DISPLAYCFG
获取显示输出位置参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_VIDEOWALLDISPLAYPOSITION 结构体: NET_DVR_VIDEOWALLDISPLAYPOSITION
设置显示输出位置参数	NET_DVR_SetDeviceConfig	命令: NET_DVR_SET_VIDEOWALLDISPLAYPOSITION 结构体: NET_DVR_VIDEOWALLDISPLAYPOSITION
获取电视墙显示输出参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_WALLOUTPUT_GET 结构体: NET_DVR_WALLOUTPUTPARAM
设置电视墙显示输出参数	NET_DVR_SetDeviceConfig	命令: NET_DVR_WALLOUTPUT_SET 结构体: NET_DVR_WALLOUTPUTPARAM
远程控制电视墙上显示编号	NET_DVR_RemoteControl	命令: NET_DVR_DISPLAY_CHANNO_CONTROL
获取虚拟 LED 信息	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_VIRTUALLED 结构体: NET_DVR_VIRTUALLED_PARAM DS-6400HD-T 不支持
设置虚拟 LED 信息	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_VIRTUALLED 结构体: NET_DVR_VIRTUALLED_PARAM DS-6400HD-T 不支持
获取 LED 区域列表	NET_DVR_GetSTDConfig	命令: NET_DVR_GET_LED_AREA_INFO_LIST 结构体: NET_DVR_LED_AREA_INFO_LIST (69UD V2.0 新增)
电视墙窗口操作		
获取电视墙窗口参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_VIDEOWALLWINDOWPOSITION 结构体: NET_DVR_VIDEOWALLWINDOWPOSITION
设置电视墙窗口参数	NET_DVR_SetDeviceConfigEx	命令: NET_DVR_SET_VIDEOWALLWINDOWPOSITION
获取电视墙窗口相关参数	NET_DVR_GetDVRConfig	命令: NET_DVR_WALLWINPARAM_GET 结构体: NET_DVR_WALLWINPARAM
设置电视墙窗口相关参数	NET_DVR_SetDVRConfig	命令: NET_DVR_WALLWINPARAM_SET 结构体: NET_DVR_WALLWINPARAM
获取窗口参数配置	NET_DVR_MatrixGetDecChanCfg	
设置窗口参数配置	NET_DVR_MatrixSetDecChanCfg	
电视墙关闭所有窗口	NET_DVR_RemoteControl	命令: NET_DVR_VIDEOWALLWINDOW_CLOSEALL
窗口置顶	NET_DVR_RemoteControl	命令: NET_DVR_SWITCH_WIN_TOP
窗口置底	NET_DVR_RemoteControl	命令: NET_DVR_SWITCH_WIN_BOTTOM
窗口解码操作		

启动动态解码	NET_DVR_MatrixStartDynamic_V41	
停止动态解码	NET_DVR_MatrixStopDynamic	
获取轮巡解码通道参数	NET_DVR_MatrixGetLoopDecChanInfo_V41	
设置轮巡解码通道参数	NET_DVR_MatrixSetLoopDecChanInfo_V41	
远程文件回放解码	NET_DVR_RemoteControl	命令: NET_DVR_DEC_PLAY_REMOTE_FILE
远程回放文件控制	NET_DVR_MatrixSetRemotePlayControl	
获取回放状态	NET_DVR_MatrixGetRemotePlayStatus	
获取解码通道开关	NET_DVR_MatrixGetDecChanEnable	
设置解码通道开关	NET_DVR_MatrixSetDecChanEnable	
获取窗口轮巡解码开关	NET_DVR_MatrixGetLoopDecChanEnable	
设置窗口轮巡解码开关	NET_DVR_MatrixSetLoopDecChanEnable	
启动被动解码	NET_DVR_MatrixStartPassiveDecode	
向被动解码窗口发送数据	NET_DVR_MatrixSendData	
停止被动解码	NET_DVR_MatrixStopPassiveDecode	
获取被动解码状态	NET_DVR_MatrixGetPassiveDecodeStatus	
被动解码播放控制	NET_DVR_MatrixPassiveDecodeControl	
获取当前窗口解码信息	NET_DVR_MatrixGetDecChanInfo_V41	
批量获取窗口解码信息	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_WIN_DEC_INFO 结构体: NET_DVR_MATRIX_DEC_CHAN_INFO_V41
获取窗口解码状态信息	NET_DVR_GetDeviceStatus	命令: NET_DVR_MATRIX_GETWINSTATUS 结构体: NET_DVR_WALL_WIN_STATUS
音频切换		DS-6400HD-S 不支持以下音频切换接口
获取音频切换参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_VW_AUDIO_CFG 结构体: NET_DVR_AUDIO_CHAN_CFG
设置音频切换参数	NET_DVR_SetDeviceConfig	命令: NET_DVR_SET_VW_AUDIO_CFG 结构体: NET_DVR_AUDIO_CHAN_CFG
场景		DS-6400HD-T 不支持以下场景接口
获取当前场景号	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_CURRENT_SCENE 结构体: 4 字节场景号
获取电视墙场景模式参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_VW_SCENE_PARAM 结构体: NET_DVR_WALLSCENECFG
设置电视墙场景模式参数	NET_DVR_SetDeviceConfig	命令: NET_DVR_SET_VW_SCENE_PARAM 结构体: NET_DVR_WALLSCENECFG
场景切换控制	NET_DVR_RemoteControl	命令: NET_DVR_SCENE_CONTROL
底图配置、上传和控制		DS-6400HD-T 不支持以下底图接口
获取底图参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_BASEMAP_CFG 结构体: NET_DVR_BASEMAP_CONTROL_CFG

设置底图参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_BASEMAP_CFG 结构体: NET_DVR_BASEMAP_CONTROL_CFG
获取底图窗口参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_BASEMAP_WIN_CFG 结构体: NET_DVR_BASEMAP_WINCFG
设置底图窗口参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_BASEMAP_WIN_CFG 结构体: NET_DVR_BASEMAP_WINCFG
获取底图图片信息	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_BASEMAP_PIC_INFO 结构体: NET_DVR_BASEMAP_PIC_INFO
底图上传	NET_DVR_PicUpload	
获取上传的进度	NET_DVR_GetPicUploadProgress	
获取上传的状态	NET_DVR_GetPicUploadState	
关闭上传句柄, 释放资源	NET_DVR_CloseUploadHandle	
删除底图	NET_DVR_RemoteControl	命令: NET_DVR_DELETE_PICTURE
LOGO 配置、上传和控制		
获取 Logo 参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_MATRIX_LOGO_CFG 结构体: NET_DVR_MATRIX_LOGO_CFG
设置 Logo 参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_MATRIX_LOGO_CFG 结构体: NET_DVR_MATRIX_LOGO_CFG
获取窗口 Logo 参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_WIN_LOGO_CFG 结构体: NET_DVR_WIN_LOGO_CFG
设置窗口 Logo 参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_WIN_LOGO_CFG 结构体: NET_DVR_WIN_LOGO_CFG
上传 LOGO	NET_DVR_UploadLogo_NEW	
删除 LOGO	NET_DVR_RemoteControl	命令: NET_DVR_DELETE_LOGO
解码 OSD 配置		
获取子窗口解码 OSD 能力集	NET_DVR_GetSTDAbility	命令: NET_DVR_GET_SUBWND_DECODE_OSD_CAP (69UD V2.0 新增)
获取所有子窗口解码 OSD 信息	NET_DVR_GetSTDConfig	命令: NET_DVR_GET_SUBWND_DECODE_OSD_ALL 结构体: NET_DVR_SUBWND_DECODE_OSD_LIST (69UD V2.0 新增)
获取子窗口解码 OSD 信息	NET_DVR_GetSTDConfig	命令: NET_DVR_GET_SUBWND_DECODE_OSD 结构体: NET_DVR_SUBWND_DECODE_OSD (69UD V2.0 新增)
设置子窗口解码 OSD 信息	NET_DVR_SetSTDConfig	命令: NET_DVR_SET_SUBWND_DECODE_OSD 结构体: NET_DVR_SUBWND_DECODE_OSD (69UD V2.0 新增)
图片回显		
开始图片回显	NET_DVR_StartPicPreview	DS-6400HD-T 不支持以下回显接口

停止图片回显	NET_DVR_StopScreenPic	
报警布防		
注册回调函数，接收设备报警消息等	NET_DVR_SetDVRMessageCallBack_V30	支持的报警信息类型有： COMM_ALARM_WALL_CONFERNECE (69UD V2.0 新增)
建立报警上传通道，获取报警等信息	NET_DVR_SetupAlarmChan_V41	
撤销报警上传通道	NET_DVR_CloseAlarmChan_V30	
获取设备运行状态		
获取设备状态信息	NET_DVR_GetDeviceStatus	命令：NET_DVR_GET_DEVICE_RUN_STATUS
设备远程维护		
设置远程升级时网络环境	NET_DVR_SetNetworkEnvironment	
远程升级设备固件	NET_DVR_Upgrade	
获取远程升级的进度	NET_DVR_GetUpgradeProgress	
获取远程升级的状态	NET_DVR_GetUpgradeState	
关闭远程升级句柄，释放资源	NET_DVR_CloseUpgradeHandle	
查找日志信息	NET_DVR_FindDVRLog_V30	
逐条获取查找到的日志信息	NET_DVR_FindNextLog_V30	
关闭日志查找，释放资源	NET_DVR_FindLogClose_V30	
重启设备	NET_DVR_RebootDVR	
恢复默认参数	NET_DVR_RestoreConfig	
导出配置文件	NET_DVR_GetConfigFile	
导入配置文件	NET_DVR_SetConfigFile	

5.3 视频综合平台

视频综合平台包括但不仅限于产品型号：DS-B20、DS-B21。除支持通用接口的功能外，还支持以下功能接口：

功能	接口	相关参数
获取设备能力集		
获取设备能力集	NET_DVR_GetDeviceAbility	能力集类型： WALL_ABILITY， DEVICE_SOFTWARE_ABILITY， DEVICE_USER_ABILITY， DEVICE_NETAPP_ABILITY， DEVICE_VIDEOPIC_ABILITY， DEVICE_SERIAL_ABILITY，

		DEVICE_ABILITY_INFO, CAM_FUSION_ABILITY(B20 V1.3 新增), OPTICAL_DEV_ACCESS_ABILITY(B20 V1.4 新增), NET_RING_ABILITY(B20 V1.4 新增)
通用参数配置		
获取设备参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_DEVICECFG_V40 结构体: NET_DVR_DEVICECFG_V40
设置设备参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_DEVICECFG_V40 结构体: NET_DVR_DEVICECFG_V40
获取时间参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_TIMECFG 结构体: NET_DVR_TIME
设置时间参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_TIMECFG 结构体: NET_DVR_TIME
获取时区和夏时制参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_ZONEANDDST 结构体: NET_DVR_ZONEANDDST
设置时区和夏时制参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_ZONEANDDST 结构体: NET_DVR_ZONEANDDST
获取网络参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_NETCFG_V30 结构体: NET_DVR_NETCFG_V30
设置网络参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_NETCFG_V30 结构体: NET_DVR_NETCFG_V30
获取异常参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_EXCEPTIONCFG_V30 结构体: NET_DVR_EXCEPTION_V30
设置异常参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_EXCEPTIONCFG_V30 结构体: NET_DVR_EXCEPTION_V30
获取用户参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_USERCFG_V40 结构体: NET_DVR_USER_V40
设置用户参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_USERCFG_V40 结构体: NET_DVR_USER_V40
获取视频综合平台报警配置	NET_DVR_GetDVRConfig	命令: NET_DVR_VIDEOPLATFORMALARMCFG_GET 结构体: NET_DVR_VIDEOPLATFORMALARMCFG
设置视频综合平台报警配置	NET_DVR_SetDVRConfig	命令: NET_DVR_VIDEOPLATFORMALARMCFG_SET 结构体: NET_DVR_VIDEOPLATFORMALARMCFG
获取端口聚合参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_ETHERNET_CHANNEL 结构体: NET_DVR_ETHERNET_CHANNEL_INFO
设置端口聚合参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_ETHERNET_CHANNEL 结构体: NET_DVR_ETHERNET_CHANNEL_INFO
获取主控板串口参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_MAINBOARD_SERIAL 结构体: NET_DVR_MAINBOARD_SERIAL_CFG

设置主控板串口参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_MAINBOARD_SERIAL 结构体: NET_DVR_MAINBOARD_SERIAL_CFG
获取通道绑定串口配置	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_CAMERACHAN_SERIALCFG 结构体: NET_DVR_CAMERACHAN_SERIALCFG (B20 V1.3 新增)
设置通道绑定串口配置	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_CAMERACHAN_SERIALCFG 结构体: NET_DVR_CAMERACHAN_SERIALCFG (B20 V1.3 新增)
获取超高清输入子系统参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_VIRTUAL_SCREEN_CFG 结构体: NET_DVR_VIRTUAL_SCREEN_CFG
设置超高清输入子系统参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_VIRTUAL_SCREEN_CFG 结构体: NET_DVR_VIRTUAL_SCREEN_CFG
获取 GBT28181 协议接入配置	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_GBT28181_ACCESS_CFG 结构体: NET_DVR_GBT28181_ACCESS_CFG
设置 GBT28181 协议接入配置	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_GBT28181_ACCESS_CFG 结构体: NET_DVR_GBT28181_ACCESS_CFG
获取 GBT28181 协议接入设备的通道信息	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_GBT28181_CHANINFO_CFG 结构体: NET_DVR_GBT28181_CHANINFO_CFG
设置 GBT28181 协议接入设备的通道信息	NET_DVR_SetDeviceConfig	命令: NET_DVR_SET_GBT28181_CHANINFO_CFG 结构体: NET_DVR_GBT28181_CHANINFO_CFG
获取 GBT28181 协议接入设备的解码通道信息	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_GBT28181_DECCHANINFO_CFG 结构体: NET_DVR_GBT28181_CHANINFO_CFG
设置 GBT28181 协议接入设备的解码通道信息	NET_DVR_SetDeviceConfig	命令: NET_DVR_SET_GBT28181_DECCHANINFO_CFG 结构体: NET_DVR_GBT28181_CHANINFO_CFG
获取设备支持的 IPC 协议列表	NET_DVR_GetIPCProtoList_V41	
信号源和设备管理		
获取所有子系统信息	NET_DVR_MatrixGetSubSystemInfo_V40	
设置子系统参数	NET_DVR_MatrixSetSubSystemInfo_V40	
获取信号源列表	NET_DVR_GetInputSignalList_V40	(B21 V1.3 新增支持)
设置输入信号源	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_INPUTSTREAMCFG_V40 结构体: NET_DVR_INPUTSTREAMCFG_V40 (B21 V1.3 新增支持)
获取显示输入参数能力集	NET_DVR_GetSTDAbility	能力集类型: NET_DVR_GET_DISPINPUT_CFG_CAP 能力集描述: InputConfigList (B21 V1.3 新增)
获取显示输入参数	NET_DVR_GetSTDConfig	命令: NET_DVR_GET_DISPINPUT_CFG 结构体: NET_DVR_DISPINPUT_CFG (B21 V1.3 新增)
获取所有显示输入参数	NET_DVR_GetSTDConfig	命令: NET_DVR_GET_DISPINPUT_CFG_LIST 结构体: NET_DVR_DISPINPUT_CFG_LIST (B21 V1.3 新增)

设置显示输入参数	NET_DVR_SetSTDConfig	命令: NET_DVR_SET_DISPINPUT_CFG 结构体: NET_DVR_DISPINPUT_CFG (B21 V1.3 新增)
电视墙操作		
获取设备显示输出号	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_VIDEOWALLDISPLAYNO 结构体: NET_DVR_DISPLAYCFG
获取显示输出位置参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_VIDEOWALLDISPLAYPOSITION 结构体: NET_DVR_VIDEOWALLDISPLAYPOSITION
设置显示输出位置参数	NET_DVR_SetDeviceConfig	命令: NET_DVR_SET_VIDEOWALLDISPLAYPOSITION 结构体: NET_DVR_VIDEOWALLDISPLAYPOSITION
获取电视墙显示输出参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_WALLOUTPUT_GET 结构体: NET_DVR_WALLOUTPUTPARAM
设置电视墙显示输出参数	NET_DVR_SetDeviceConfig	命令: NET_DVR_WALLOUTPUT_SET 结构体: NET_DVR_WALLOUTPUTPARAM
远程控制电视墙上显示编号	NET_DVR_RemoteControl	命令: NET_DVR_DISPLAY_CHANNO_CONTROL
获取虚拟 LED 信息	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_VIRTUALLED 结构体: NET_DVR_VIRTUALLED_PARAM
设置虚拟 LED 信息	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_VIRTUALLED 结构体: NET_DVR_VIRTUALLED_PARAM
获取电视墙全局参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_WALL_CFG 结构体: NET_DVR_WALL_CFG
设置电视墙全局参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_WALL_CFG 结构体: NET_DVR_WALL_CFG
屏幕控制	NET_DVR_RemoteControl	命令: NET_DVR_CONTROL_SCREEN (B20 V1.4 新增支持)
获取 LED 区域列表	NET_DVR_GetSTDConfig	命令: NET_DVR_GET_LED_AREA_INFO_LIST 结构体: NET_DVR_LED_AREA_INFO_LIST (B21 V1.3 新增支持)
电视墙窗口操作		
获取电视墙窗口参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_VIDEOWALLWINDOWPOSITION 结构体: NET_DVR_VIDEOWALLWINDOWPOSITION
设置电视墙窗口参数	NET_DVR_SetDeviceConfigEx	命令: NET_DVR_SET_VIDEOWALLWINDOWPOSITION
获取电视墙窗口相关参数	NET_DVR_GetDVRConfig	命令: NET_DVR_WALLWINPARAM_GET 结构体: NET_DVR_WALLWINPARAM
设置电视墙窗口相关参数	NET_DVR_SetDVRConfig	命令: NET_DVR_WALLWINPARAM_SET 结构体: NET_DVR_WALLWINPARAM
获取窗口参数配置	NET_DVR_MatrixGetDecChanCfg	
设置窗口参数配置	NET_DVR_MatrixSetDecChanCfg	
电视墙关闭所有窗口	NET_DVR_RemoteControl	命令: NET_DVR_VIDEOWALLWINDOW_CLOSEALL

窗口置顶	NET_DVR_RemoteControl	命令: NET_DVR_SWITCH_WIN_TOP
窗口置底	NET_DVR_RemoteControl	命令: NET_DVR_SWITCH_WIN_BOTTOM
窗口解码操作		
启动动态解码	NET_DVR_MatrixStartDynamic_V41	
停止动态解码	NET_DVR_MatrixStopDynamic	
获取轮巡解码通道参数	NET_DVR_MatrixGetLoopDecChanInfo_V41	
设置轮巡解码通道参数	NET_DVR_MatrixSetLoopDecChanInfo_V41	
远程文件回放解码	NET_DVR_RemoteControl	命令: NET_DVR_DEC_PLAY_REMOTE_FILE
远程回放文件控制	NET_DVR_MatrixSetRemotePlayControl	
获取回放状态	NET_DVR_MatrixGetRemotePlayStatus	
获取解码通道开关	NET_DVR_MatrixGetDecChanEnable	
设置解码通道开关	NET_DVR_MatrixSetDecChanEnable	
获取窗口轮巡解码开关	NET_DVR_MatrixGetLoopDecChanEnable	
设置窗口轮巡解码开关	NET_DVR_MatrixSetLoopDecChanEnable	
启动被动解码	NET_DVR_MatrixStartPassiveDecode	
向被动解码窗口发送数据	NET_DVR_MatrixSendData	
批量动态解码	NET_DVR_SetDeviceConfig	命令: NET_DVR_DYNAMIC_DECODE_BATCH 结构体: NET_DVR_PU_STREAM_CFG_V41 (B20 V1.4 新增支持)
批量获取解码通道开关	NET_DVR_GetDeviceConfig	命令: NET_DVR_DECSWITCH_GET_BATCH (B20 V1.4 新增支持)
批量设置解码通道开关	NET_DVR_SetDeviceConfig	命令: NET_DVR_DECSWITCH_SET_BATCH (B20 V1.4 新增支持)
停止被动解码	NET_DVR_MatrixStopPassiveDecode	
获取被动解码状态	NET_DVR_MatrixGetPassiveDecodeStatus	
被动解码播放控制	NET_DVR_MatrixPassiveDecodeControl	
获取当前窗口解码信息	NET_DVR_MatrixGetDecChanInfo_V41	
批量获取窗口解码信息	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_WIN_DEC_INFO 结构体: NET_DVR_MATRIX_DEC_CHAN_INFO_V41
获取电视墙窗口对应资源信息	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_WINASSOCIATEDDEVINFO 结构体: NET_DVR_ASSOCIATEDDEV_CHAN_INFO
获取窗口解码状态信息	NET_DVR_GetDeviceStatus	命令: NET_DVR_MATRIX_GETWINSTATUS 结构体: NET_DVR_WALL_WIN_STATUS
窗口 JPEG 抓图, 保存成文件	NET_DVR_CaptureJPEGPicture	
窗口 JPEG 抓图, 保存在缓冲区	NET_DVR_CaptureJPEGPicture_NEW	
大屏电子放大	NET_DVR_ScreenZoomIn	(B20 V1.3 新增支持)

音频切换		
获取音频切换参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_VW_AUDIO_CFG 结构体: NET_DVR_AUDIO_CHAN_CFG
设置音频切换参数	NET_DVR_SetDeviceConfig	命令: NET_DVR_SET_VW_AUDIO_CFG 结构体: NET_DVR_AUDIO_CHAN_CFG
场景		
获取当前场景号	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_CURRENT_SCENE 结构体: 4 字节场景号
获取电视墙场景模式参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_VW_SCENE_PARAM 结构体: NET_DVR_WALLSCENECFG
设置电视墙场景模式参数	NET_DVR_SetDeviceConfig	命令: NET_DVR_SET_VW_SCENE_PARAM 结构体: NET_DVR_WALLSCENECFG
场景切换控制	NET_DVR_RemoteControl	命令: NET_DVR_SCENE_CONTROL
导入场景配置文件	NET_DVR_UploadFile_V40 NET_DVR_GetUploadState NET_DVR_UploadClose	上传文件类型: SCENE_CONFIGURATION_FILE
导出场景配置文件	NET_DVR_StartDownload NET_DVR_GetDownloadState NET_DVR_StopDownload	下载文件类型: NET_SDK_DOWNLOAD_SCENE_CONFIGURATION_FILE
预案		
获取预案列表	NET_DVR_GetPlanList	
设置预案	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_PLAN 结构体: NET_DVR_PLAN_CFG
预案控制	NET_DVR_RemoteControl	命令: NET_DVR_CTRL_PLAN
获取正在执行预案信息	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_PLAYING_PLAN (B20 V1.4 新增支持)
底图配置、上传和控制		
获取底图参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_BASEMAP_CFG 结构体: NET_DVR_BASEMAP_CONTROL_CFG
设置底图参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_BASEMAP_CFG 结构体: NET_DVR_BASEMAP_CONTROL_CFG
获取底图窗口参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_BASEMAP_WIN_CFG 结构体: NET_DVR_BASEMAP_WINCFG
设置底图窗口参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_BASEMAP_WIN_CFG 结构体: NET_DVR_BASEMAP_WINCFG
获取底图图片信息	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_BASEMAP_PIC_INFO 结构体: NET_DVR_BASEMAP_PIC_INFO
底图上传	NET_DVR_PicUpload	

获取上传的进度	NET_DVR_GetPicUploadProgress	
获取上传的状态	NET_DVR_GetPicUploadState	
关闭上传句柄，释放资源	NET_DVR_CloseUploadHandle	
删除底图	NET_DVR_RemoteControl	命令: NET_DVR_DELETE_PICTURE
LOGO 配置、上传和控制		
获取 Logo 参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_MATRIX_LOGO_CFG 结构体: NET_DVR_MATRIX_LOGO_CFG
设置 Logo 参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_MATRIX_LOGO_CFG 结构体: NET_DVR_MATRIX_LOGO_CFG
获取窗口 Logo 参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_WIN_LOGO_CFG 结构体: NET_DVR_WIN_LOGO_CFG
设置窗口 Logo 参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_WIN_LOGO_CFG 结构体: NET_DVR_WIN_LOGO_CFG
上传 LOGO	NET_DVR_UploadLogo_NEW	
下载 LOGO	NET_DVR_DownloadLogo	(B20 V1.3 新增)
删除 LOGO	NET_DVR_RemoteControl	命令: NET_DVR_DELETE_LOGO
图片回显		
开始图片回显	NET_DVR_StartPicPreview	
停止图片回显	NET_DVR_StopScreenPic	
图像微调		
图像微调	NET_DVR_MatrixPicAdjust	
透明通道		
建立透明通道	NET_DVR_SerialStart	
通过透明通道向设备串口发送数据	NET_DVR_SerialSend	
断开透明通道	NET_DVR_SerialStop	
云台控制		
云台控制	NET_DVR_PTZControlWithSpeed_Other	(B20 V1.3 新增支持通过主控板控制云台)
3 维带速度云台控制	NET_DVR_RemoteControl	命令: NET_DVR_PTZ_3D_SPEED(B10 V1.3 新增)
获取设备支持云台协议列表	NET_DVR_GetPTZProtocol_Ex	(B20 V1.3 新增支持)
相机拼接配置		(B20 V1.3 新增支持)
获取流来源信息	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_STREAM_SRC_INFO 结构体: NET_DVR_STREAM_SRC_INFO
获取所有流信息	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_ALLSTREAM_SRC_INFO 结构体: NET_DVR_STREAM_SRC_CFG
设置流来源信息	NET_DVR_SetDeviceConfig	命令: NET_DVR_SET_STREAM_SRC_INFO 结构体: NET_DVR_STREAM_SRC_INFO

获取图像融合规模	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_FUSION_SCALE 结构体: NET_DVR_FUSION_SCALE
设置图像融合规模	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_FUSION_SCALE 结构体: NET_DVR_FUSION_SCALE
获取图像融合参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_FUSION_CFG 结构体: NET_DVR_FUSION_CFG
设置图像融合参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_FUSION_CFG 结构体: NET_DVR_FUSION_CFG
信号源拼接		
获取信号源绑定配置	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_SIGNAL_JOINT 结构体: NET_DVR_SIGNAL_JOINT_CFG (B20 V1.4 新增支持)
设置信号源绑定配置	NET_DVR_SetDeviceConfigEx	命令: NET_DVR_SET_SIGNAL_JOINT (B20 V1.4 新增支持)
获取所有信号源绑定配置	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_ALL_SIGNAL_JOINT (B20 V1.4 新增支持)
删除拼接信号源	NET_DVR_RemoteControl	命令: NET_DVR_DEL_SIGNAL_JOINT (B20 V1.4 新增)
光端机相关配置		
获取光纤板环网配置	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_NS_RING_CFG 结构体: NET_DVR_NS_RING_CFG (B20 V1.4 新增)
设置光纤板环网配置	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_NS_RING_CFG 结构体: NET_DVR_NS_RING_CFG (B20 V1.4 新增)
获取光纤板环网状态	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_NS_RING_STATUS 结构体: NET_DVR_NS_RING_STATUS (B20 V1.4 新增)
获取光口信息	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_OPTICAL_PORT_INFO 结构体: NET_DVR_OPTICAL_PORT_INFO (B20 V1.4 新增)
设置光口信息	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_OPTICAL_PORT_INFO 结构体: NET_DVR_OPTICAL_PORT_INFO (B20 V1.4 新增)
获取编码通道关联光口输入源参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_OPTICAL_CHAN_RELATE_CFG 结构体: NET_DVR_OPTICAL_CHAN_RELATE_CFG (B20 V1.4 新增)
设置编码通道关联光口输入源参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_OPTICAL_CHAN_RELATE_CFG 结构体: NET_DVR_OPTICAL_CHAN_RELATE_CFG (B20 V1.4 新增)
报警布防		
注册回调函数, 接收设备报警消	NET_DVR_SetDVRMessageCallBack V30	支持的报警信息类型有:

息等		COMM_ALARM_VIDEOPLATFORM
建立报警上传通道, 获取报警等信息	NET_DVR_SetupAlarmChan_V41	
撤销报警上传通道	NET_DVR_CloseAlarmChan_V30	
报警监听		
启动监听	NET_DVR_StartListen_V30	支持的报警信息类型有: COMM_ALARM_VIDEOPLATFORM
停止监听	NET_DVR_StopListen_V30	
获取设备运行状态		
获取设备状态信息	NET_DVR_GetDeviceStatus	命令: NET_DVR_GET_DEVICE_RUN_STATUS 结构体: NET_DVR_DEVICE_RUN_STATUS
获取视频综合平台状态	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_MATRIX_STATUS 结构体: NET_DVR_MATRIX_STATUS (B20 V1.3 新增)
EDID 文件管理		
上传 EDID 文件	NET_DVR_UploadFile_V40 NET_DVR_GetUploadState NET_DVR_UploadClose	上传文件类型: UPLOAD_EDID_CFG_FILE (B21 V1.3 新增)
获取 EDID 文件信息配置能力集	NET_DVR_GetSTDAbility	能力集类型: NET_DVR_GET_EDID_CFG_FILE_INFO_CAP (B21 V1.3 新增)
获取 EDID 配置文件信息	NET_DVR_GetSTDConfig	命令: NET_DVR_GET_EDID_CFG_FILE_INFO 结构体: NET_DVR_EDID_FILE_INFO (B21 V1.3 新增)
获取所有 EDID 配置文件信息	NET_DVR_GetSTDConfig	命令: NET_DVR_GET_EDID_CFG_FILE_INFO_LIST 结构体: NET_DVR_EDID_FILE_INFO_LIST (B21 V1.3 新增)
设置 EDID 配置文件信息	NET_DVR_SetSTDConfig	命令: NET_DVR_SET_EDID_CFG_FILE_INFO 结构体: NET_DVR_EDID_FILE_INFO (B21 V1.3 新增)
删除 EDID 配置文件信息	NET_DVR_STDControl	命令: NET_DVR_DEL_EDID_CFG_FILE_INFO (B21 V1.3 新增)
设备远程维护		
设置远程升级时网络环境	NET_DVR_SetNetworkEnvironment	
远程升级设备固件	NET_DVR_Upgrade	
获取远程升级的进度	NET_DVR_GetUpgradeProgress	
获取远程升级的状态	NET_DVR_GetUpgradeState	
关闭远程升级句柄, 释放资源	NET_DVR_CloseUpgradeHandle	

查找日志信息	NET_DVR_FindDVRLog_Matrix	
逐条获取查找到的日志信息	NET_DVR_FindNextLog_MATRIX	
关闭日志查找，释放资源	NET_DVR_FindLogClose_V30	
重启设备	NET_DVR_RebootDVR	
恢复默认参数	NET_DVR_RestoreConfig	
配置文件导入导出方式一		
导出配置文件	NET_DVR_MatrixGetConfigFile	
导入配置文件	NET_DVR_MatrixSetConfigFile	
配置文件导入导出方式二		
导入配置文件	NET_DVR_UploadFile NET_DVR_GetUploadState NET_DVR_UploadClose	上传文件类型：CONFIGURATION_FILE
导出配置文件	NET_DVR_StartDownload NET_DVR_GetDownloadState NET_DVR_StopDownload	下载文件类型： NET_SDK_DOWNLOAD_CONFIGURATION_FILE

5.4 集中式多屏控制器

集中式多屏控制器包括但不限于产品型号：DS-C10S。除支持通用接口的功能外，还支持以下功能接口：

功能	接口	相关参数
获取设备能力集		
获取设备软硬件能力	NET_DVR_GetDeviceAbility	能力集类型：DEVICE_SOFTHARDWARE_ABILITY， 能力集：软硬件能力 XML 描述
获取设备网络应用参数能力	NET_DVR_GetDeviceAbility	能力集类型：DEVICE_NETAPP_ABILITY， 能力集：网络应用参数能力 XML 描述
获取设备用户管理参数能力	NET_DVR_GetDeviceAbility	能力集类型：DEVICE_USER_ABILITY， 能力集：用户管理参数能力 XML 描述
获取电视墙能力	NET_DVR_GetDeviceAbility	能力集类型：WALL_ABILITY， 能力集：电视墙能力 XML 描述
通用参数配置		
获取设备参数	NET_DVR_GetDVRConfig	命令：NET_DVR_GET_DEVICECFG_V40 结构体： NET_DVR_DEVICECFG_V40
设置设备参数	NET_DVR_SetDVRConfig	命令：NET_DVR_SET_DEVICECFG_V40 结构体： NET_DVR_DEVICECFG_V40
获取时间参数	NET_DVR_GetDVRConfig	命令：NET_DVR_GET_TIMECFG 结构体： NET_DVR_TIME
设置时间参数	NET_DVR_SetDVRConfig	命令：NET_DVR_SET_TIMECFG

		结构体: NET_DVR_TIME
获取时区和夏时制参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_ZONEANDDST 结构体: NET_DVR_ZONEANDDST
设置时区和夏时制参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_ZONEANDDST 结构体: NET_DVR_ZONEANDDST
获取网络参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_NETCFG_V30 结构体: NET_DVR_NETCFG_V30
设置网络参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_NETCFG_V30 结构体: NET_DVR_NETCFG_V30
获取异常参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_EXCEPTIONCFG_V30 结构体: NET_DVR_EXCEPTION_V30
设置异常参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_EXCEPTIONCFG_V30 结构体: NET_DVR_EXCEPTION_V30
获取用户信息	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_VCS_USER_CFG 结构体: NET_DVR_VCS_USER_INFO
设置用户信息	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_VCS_USER_CFG 结构体: NET_DVR_VCS_USER_INFO
获取当前使用串口	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_USING_SERIALPORT 结构体: NET_DVR_USING_SERIALPORT (V2.0 新增支持)
设置当前使用串口	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_USING_SERIALPORT 结构体: NET_DVR_USING_SERIALPORT (V2.0 新增支持)
获取主控板串口参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_MAINBOARD_SERIAL 结构体: NET_DVR_MAINBOARD_SERIAL_CFG (V2.0 新增支持)
设置主控板串口参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_MAINBOARD_SERIAL 结构体: NET_DVR_MAINBOARD_SERIAL_CFG (V2.0 新增支持)
信号源管理		
获取信号源列表	NET_DVR_GetInputSignalList_V40	(V2.0 扩展)
设置输入信号源	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_INPUTSTREAMCFG_V40 结构体: NET_DVR_INPUTSTREAMCFG_V40 (V2.0 扩展)
获取矩阵信息	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_EXTERNAL_MATRIX_CFG 结构体: NET_DVR_EXTERNAL_MATRIX_CFG
设置矩阵信息	NET_DVR_SetDeviceConfig	命令: NET_DVR_SET_EXTERNAL_MATRIX_CFG 结构体: NET_DVR_EXTERNAL_MATRIX_CFG
获取解码板网络参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_DEV_NETINFO

		结构体: NET_DVR_NETPARAM
设置解码板网络参数	NET_DVR_SetDeviceConfig	命令: NET_DVR_SET_DEV_NETINFO 结构体: NET_DVR_NETPARAM
获取信号源绑定配置	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_SIGNAL_JOINT 结构体: NET_DVR_SIGNAL_JOINT_CFG (V2.0 新增)
设置信号源绑定配置	NET_DVR_SetDeviceConfigEx	命令: NET_DVR_SET_SIGNAL_JOINT (V2.0 新增)
获取所有信号源绑定配置	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_ALL_SIGNAL_JOINT 结构体: NET_DVR_SIGNAL_JOINT_CFG (V2.0 新增)
信号源裁剪控制	NET_DVR_RemoteControl	命令: NET_DVR_SIGNAL_CUT (V2.0 新增)
获取信号源裁剪参数	NET_DVR_GetDVRConfig	命令: NET_DVR_SIGNAL_CUTPARAM_GET 结构体: NET_DVR_SIGNAL_CUTPARAM (V2.0 新增)
子板信息		
获取设备子板信息	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_SUBBOARD_INFO 结构体: NET_DVR_DEVICE_SUBBOARD_INFO
获取子板异常信息	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_SUBBOARD_EXCEPTION 结构体: NET_DVR_DEVICE_BOARD_EXCEPINFO
屏幕服务器登录参数配置		
屏幕服务器登录参数配置	NET_DVR_STDXMLConfig	透传 XML 协议, 分别通过 GET、PUT、POST、DELETE 命令获取和设置、添加、删除参数以及获取能力 (V2.1 新增)
输入板参数配置		
获取输入板配置信息列表	NET_DVR_GetSTDConfig	命令: NET_DVR_GET_INPUT_BOARD_CFG_LIST 结构体: NET_DVR_INPUT_BOARD_CFG_LIST (V2.1 新增)
获取输入板配置信息	NET_DVR_GetSTDConfig	命令: NET_DVR_GET_INPUT_BOARD_CFG 结构体: NET_DVR_INPUT_BOARD_CFG (V2.1 新增)
设置输入板配置信息	NET_DVR_SetSTDConfig	命令: NET_DVR_SET_INPUT_BOARD_CFG 结构体: NET_DVR_INPUT_BOARD_CFG (V2.1 新增)
IPC 协议列表		
获取 IPC 协议列表	NET_DVR_GetIPCProtoList_V41	
电视墙操作		
获取电视墙拼接模式	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_VIDEOWALLDISPLAYMODE 结构体: NET_DVR_VIDEOWALLDISPLAYMODE
设置电视墙拼接模式	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_VIDEOWALLDISPLAYMODE 结构体: NET_DVR_VIDEOWALLDISPLAYMODE
获取设备显示输出号	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_VIDEOWALLDISPLAYNO 结构体: NET_DVR_DISPLAYCFG

获取显示输出位置参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_VIDEOWALLDISPLAYPOSITION 结构体: NET_DVR_VIDEOWALLDISPLAYPOSITION
设置显示输出位置参数	NET_DVR_SetDeviceConfig	命令: NET_DVR_SET_VIDEOWALLDISPLAYPOSITION 结构体: NET_DVR_VIDEOWALLDISPLAYPOSITION
获取电视墙显示输出参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_WALLOUTPUT_GET 结构体: NET_DVR_WALLOUTPUTPARAM
设置电视墙显示输出参数	NET_DVR_SetDeviceConfig	命令: NET_DVR_WALLOUTPUT_SET 结构体: NET_DVR_WALLOUTPUTPARAM
远程控制电视墙上显示编号	NET_DVR_RemoteControl	命令: NET_DVR_DISPLAY_CHANNO_CONTROL
屏幕控制	NET_DVR_RemoteControl	命令: NET_DVR_CONTROL_SCREEN
解除电视墙输出接口绑定	NET_DVR_RemoteControl	命令: NET_DVR_RESET_VIDEOWALLDISPLAYPOSITION (V2.1 新增)
获取虚拟 LED 信息	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_VIRTUALLED 结构体: NET_DVR_VIRTUALLED_PARAM
设置虚拟 LED 信息	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_VIRTUALLED 结构体: NET_DVR_VIRTUALLED_PARAM
电视墙窗口操作		
获取电视墙窗口参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_VIDEOWALLWINDOWPOSITION 结构体: NET_DVR_VIDEOWALLWINDOWPOSITION
设置电视墙窗口参数	NET_DVR_SetDeviceConfigEx	命令: NET_DVR_SET_VIDEOWALLWINDOWPOSITION
获取电视墙窗口相关参数	NET_DVR_GetDVRConfig	命令: NET_DVR_WALLWINPARAM_GET 结构体: NET_DVR_WALLWINPARAM (V1.3 新增支持窗口分屏模式)
设置电视墙窗口相关参数	NET_DVR_SetDVRConfig	命令: NET_DVR_WALLWINPARAM_SET 结构体: NET_DVR_WALLWINPARAM (V1.3 新增支持窗口分屏模式)
电视墙关闭所有窗口	NET_DVR_RemoteControl	命令: NET_DVR_VIDEOWALLWINDOW_CLOSEALL
窗口置顶	NET_DVR_RemoteControl	命令: NET_DVR_SWITCH_WIN_TOP
窗口置底	NET_DVR_RemoteControl	命令: NET_DVR_SWITCH_WIN_BOTTOM
窗口解码操作		
启动动态解码	NET_DVR_MatrixStartDynamic_V41	
停止动态解码	NET_DVR_MatrixStopDynamic	
获取解码通道开关	NET_DVR_MatrixGetDecChanEnable	
设置解码通道开关	NET_DVR_MatrixSetDecChanEnable	
批量动态解码	NET_DVR_SetDeviceConfig	命令: NET_DVR_DYNAMIC_DECODE_BATCH 结构体: NET_DVR_PU_STREAM_CFG_V41 (V2.0 新增)
批量获取解码通道开关	NET_DVR_GetDeviceConfig	命令: NET_DVR_DEC SWITCH_GET_BATCH(V2.0 新增)

批量设置解码通道开关	NET_DVR_SetDeviceConfig	命令: NET_DVR_DECSWITCH_SET_BATCH(V2.0 新增)
启动被动解码	NET_DVR_MatrixStartPassiveDecode	只支持文件流
向被动解码窗口发送数据	NET_DVR_MatrixSendData	
停止被动解码	NET_DVR_MatrixStopPassiveDecode	
获取被动解码状态	NET_DVR_MatrixGetPassiveDecodeStatus	
被动解码播放控制	NET_DVR_MatrixPassiveDecodeControl	
获取当前窗口解码信息	NET_DVR_MatrixGetDecChanInfo_V41	
批量获取窗口解码信息	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_WIN_DEC_INFO 结构体: NET_DVR_MATRIX_DEC_CHAN_INFO_V41
获取窗口解码状态信息	NET_DVR_GetDeviceStatus	命令: NET_DVR_MATRIX_GETWINSTATUS 结构体: NET_DVR_WALL_WIN_STATUS
大屏电子放大	NET_DVR_ScreenZoomIn	
场景		
获取当前场景号	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_CURRENT_SCENE 结构体: 4 字节场景号
获取电视墙场景模式参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_VW_SCENE_PARAM 结构体: NET_DVR_WALLSCENECFG
设置电视墙场景模式参数	NET_DVR_SetDeviceConfig	命令: NET_DVR_SET_VW_SCENE_PARAM 结构体: NET_DVR_WALLSCENECFG
场景切换控制	NET_DVR_RemoteControl	命令: NET_DVR_SCENE_CONTROL
预案		
获取预案列表	NET_DVR_GetPlanList	
设置预案	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_PLAN 结构体: NET_DVR_PLAN_CFG
预案控制	NET_DVR_RemoteControl	命令: NET_DVR_CTRL_PLAN
获取正在执行预案信息	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_PLAYING_PLAN (V2.0 新增)
底图配置、上传和控制		
获取底图参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_BASEMAP_CFG 结构体: NET_DVR_BASEMAP_CONTROL_CFG
设置底图参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_BASEMAP_CFG 结构体: NET_DVR_BASEMAP_CONTROL_CFG
获取底图窗口参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_BASEMAP_WIN_CFG 结构体: NET_DVR_BASEMAP_WINCFG
设置底图窗口参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_BASEMAP_WIN_CFG 结构体: NET_DVR_BASEMAP_WINCFG
获取底图图片信息	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_BASEMAP_PIC_INFO 结构体: NET_DVR_BASEMAP_PIC_INFO

底图上传	NET_DVR_PicUpload	
获取上传的进度	NET_DVR_GetPicUploadProgress	
获取上传的状态	NET_DVR_GetPicUploadState	
关闭上传句柄，释放资源	NET_DVR_CloseUploadHandle	
图片回显		
开始图片回显	NET_DVR_StartPicPreview	
停止图片回显	NET_DVR_StopScreenPic	
图像微调		
图像微调	NET_DVR_MatrixPicAdjust	
云台控制		
获取设备支持云台协议列表	NET_DVR_GetPTZProtocol_Ex	
获取输入源反向云台控制配置	NET_DVR_GetDVRConfig	命令: NET_DVR_PTZCFG_INPUTSTREAM_GET 结构体: NET_DVR_INPUTSTREAM_PTZCFG_PARAM (V2.0 新增)
设置输入源反向云台控制配置	NET_DVR_SetDVRConfig	命令: NET_DVR_PTZCFG_INPUTSTREAM_SET 结构体: NET_DVR_INPUTSTREAM_PTZCFG_PARAM (V2.0 新增)
云台控制	NET_DVR_PTZControlWithSpeed_Other	(V2.0 新增支持)
报警布防		
注册回调函数，接收设备报警消息等	NET_DVR_SetDVRMessageCallBack_V30	支持的报警信息类型有: COMM_SCREEN_ALARM
建立报警上传通道，获取报警等信息	NET_DVR_SetupAlarmChan_V41	
撤销报警上传通道	NET_DVR_CloseAlarmChan_V30	
获取设备运行状态		
获取设备状态信息	NET_DVR_GetDeviceStatus	命令: NET_DVR_GET_DEVICE_RUN_STATUS
设备远程维护		
设置远程升级时网络环境	NET_DVR_SetNetworkEnvironment	
远程升级设备固件	NET_DVR_Upgrade	
获取远程升级的进度	NET_DVR_GetUpgradeProgress	
获取远程升级的状态	NET_DVR_GetUpgradeState	
关闭远程升级句柄，释放资源	NET_DVR_CloseUpgradeHandle	
查找日志信息	NET_DVR_FindDVRLog_V30	
逐条获取查找到的日志信息	NET_DVR_FindNextLog_V30	
关闭日志查找，释放资源	NET_DVR_FindLogClose_V30	
重启设备	NET_DVR_RebootDVR	

恢复默认参数	NET_DVR_RestoreConfig	
导出配置文件	NET_DVR_GetConfigFile	
导入配置文件	NET_DVR_SetConfigFile	

5.5 分布式多屏控制器

分布式多屏控制器包括但不限于产品型号：DS-C10N。除支持通用接口的功能外，还支持以下功能接口：

功能	接口	相关参数
获取设备能力集		
获取设备软硬件能力	NET_DVR_GetDeviceAbility	能力集类型：DEVICE_SOFTHARDWARE_ABILITY， 能力集：软硬件能力 XML 描述
获取设备网络应用参数能力	NET_DVR_GetDeviceAbility	能力集类型：DEVICE_NETAPP_ABILITY， 能力集：网络应用参数能力 XML 描述
获取设备用户管理参数能力	NET_DVR_GetDeviceAbility	能力集类型：DEVICE_USER_ABILITY， 能力集：用户管理参数能力 XML 描述
获取电视墙能力	NET_DVR_GetDeviceAbility	能力集类型：WALL_ABILITY， 能力集：电视墙能力 XML 描述
通用参数配置		
获取设备参数	NET_DVR_GetDVRConfig	命令：NET_DVR_GET_DEVICECFG_V40 结构体： NET_DVR_DEVICECFG_V40
设置设备参数	NET_DVR_SetDVRConfig	命令：NET_DVR_SET_DEVICECFG_V40 结构体： NET_DVR_DEVICECFG_V40
获取时间参数	NET_DVR_GetDVRConfig	命令：NET_DVR_GET_TIMECFG 结构体： NET_DVR_TIME
设置时间参数	NET_DVR_SetDVRConfig	命令：NET_DVR_SET_TIMECFG 结构体： NET_DVR_TIME
获取时区和夏时制参数	NET_DVR_GetDVRConfig	命令：NET_DVR_GET_ZONEANDDST 结构体： NET_DVR_ZONEANDDST
设置时区和夏时制参数	NET_DVR_SetDVRConfig	命令：NET_DVR_SET_ZONEANDDST 结构体： NET_DVR_ZONEANDDST
获取网络参数	NET_DVR_GetDVRConfig	命令：NET_DVR_GET_NETCFG_V30 结构体： NET_DVR_NETCFG_V30
设置网络参数	NET_DVR_SetDVRConfig	命令：NET_DVR_SET_NETCFG_V30 结构体： NET_DVR_NETCFG_V30
获取异常参数	NET_DVR_GetDVRConfig	命令：NET_DVR_GET_EXCEPTIONCFG_V30 结构体： NET_DVR_EXCEPTION_V30
设置异常参数	NET_DVR_SetDVRConfig	命令：NET_DVR_SET_EXCEPTIONCFG_V30

		结构体: NET_DVR_EXCEPTION_V30
获取用户信息	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_VCS_USER_CFG 结构体: NET_DVR_VCS_USER_INFO
设置用户信息	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_VCS_USER_CFG 结构体: NET_DVR_VCS_USER_INFO
获取大屏串口信息	NET_DVR_GetDVRConfig	命令: NET_DVR_BIGSCREEN_GETSERIAL 结构体: NET_DVR_SERIAL_CONTROL
获取当前使用串口	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_USING_SERIALPORT 结构体: NET_DVR_USING_SERIALPORT
设置当前使用串口	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_USING_SERIALPORT 结构体: NET_DVR_USING_SERIALPORT
信号源管理		
获取信号源列表	NET_DVR_GetInputSignalList_V40	
设置输入信号源	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_INPUTSTREAMCFG_V40 结构体: NET_DVR_INPUTSTREAMCFG_V40
获取分布式子设备信息列表	NET_DVR_GetDevList	
获取单个设备信息	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_DEV_BASEINFO 结构体: NET_DVR_DEV_BASE_INFO
设置单个设备信息	NET_DVR_SetDeviceConfig	命令: NET_DVR_SET_DEV_BASEINFO 结构体: NET_DVR_DEV_BASE_INFO
获取子设备网络参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_DEV_NETINFO 结构体: NET_DVR_NETPARAM
设置子设备网络参数	NET_DVR_SetDeviceConfig	命令: NET_DVR_SET_DEV_NETINFO 结构体: NET_DVR_NETPARAM
电视墙操作		
获取设备显示输出号	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_VIDEOWALLDISPLAYNO 结构体: NET_DVR_DISPLAYCFG
获取显示输出位置参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_VIDEOWALLDISPLAYPOSITION 结构体: NET_DVR_VIDEOWALLDISPLAYPOSITION
设置显示输出位置参数	NET_DVR_SetDeviceConfig	命令: NET_DVR_SET_VIDEOWALLDISPLAYPOSITION 结构体: NET_DVR_VIDEOWALLDISPLAYPOSITION
获取电视墙显示输出参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_WALLOUTPUT_GET 结构体: NET_DVR_WALLOUTPUTPARAM
设置电视墙显示输出参数	NET_DVR_SetDeviceConfig	命令: NET_DVR_WALLOUTPUT_SET 结构体: NET_DVR_WALLOUTPUTPARAM
远程控制电视墙上显示编号	NET_DVR_RemoteControl	命令: NET_DVR_DISPLAY_CHANNO_CONTROL
获取虚拟 LED 信息	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_VIRTUALLED 结构体: NET_DVR_VIRTUALLED_PARAM

设置虚拟 LED 信息	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_VIRTUALLED 结构体: NET_DVR_VIRTUALLED_PARAM
电视墙窗口操作		
获取电视墙窗口参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_VIDEOWALLWINDOWPOSITION 结构体: NET_DVR_VIDEOWALLWINDOWPOSITION
设置电视墙窗口参数	NET_DVR_SetDeviceConfigEx	命令: NET_DVR_SET_VIDEOWALLWINDOWPOSITION
获取电视墙窗口相关参数	NET_DVR_GetDVRConfig	命令: NET_DVR_WALLWINPARAM_GET 结构体: NET_DVR_WALLWINPARAM
设置电视墙窗口相关参数	NET_DVR_SetDVRConfig	命令: NET_DVR_WALLWINPARAM_SET 结构体: NET_DVR_WALLWINPARAM
电视墙关闭所有窗口	NET_DVR_RemoteControl	命令: NET_DVR_VIDEOWALLWINDOW_CLOSEALL
窗口置顶	NET_DVR_RemoteControl	命令: NET_DVR_SWITCH_WIN_TOP
窗口置底	NET_DVR_RemoteControl	命令: NET_DVR_SWITCH_WIN_BOTTOM
窗口解码操作		
启动动态解码	NET_DVR_MatrixStartDynamic_V41	
停止动态解码	NET_DVR_MatrixStopDynamic	
获取解码通道开关	NET_DVR_MatrixGetDecChanEnable	
设置解码通道开关	NET_DVR_MatrixSetDecChanEnable	
获取当前窗口解码信息	NET_DVR_MatrixGetDecChanInfo_V41	
批量获取窗口解码信息	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_WIN_DEC_INFO 结构体: NET_DVR_MATRIX_DEC_CHAN_INFO_V41
获取窗口解码状态信息	NET_DVR_GetDeviceStatus	命令: NET_DVR_MATRIX_GETWINSTATUS 结构体: NET_DVR_WALL_WIN_STATUS
场景		
获取当前场景号	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_CURRENT_SCENE 结构体: 4 字节场景号
获取电视墙场景模式参数	NET_DVR_GetDeviceConfig	命令: NET_DVR_GET_VW_SCENE_PARAM 结构体: NET_DVR_WALLSCENECFG
设置电视墙场景模式参数	NET_DVR_SetDeviceConfig	命令: NET_DVR_SET_VW_SCENE_PARAM 结构体: NET_DVR_WALLSCENECFG
场景切换控制	NET_DVR_RemoteControl	命令: NET_DVR_SCENE_CONTROL
预案		
获取预案列表	NET_DVR_GetPlanList	
设置预案	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_PLAN 结构体: NET_DVR_PLAN_CFG
预案控制	NET_DVR_RemoteControl	命令: NET_DVR_CTRL_PLAN
底图配置、上传和控制		

获取底图参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_BASEMAP_CFG 结构体: NET_DVR_BASEMAP_CONTROL_CFG
设置底图参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_BASEMAP_CFG 结构体: NET_DVR_BASEMAP_CONTROL_CFG
获取底图窗口参数	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_BASEMAP_WIN_CFG 结构体: NET_DVR_BASEMAP_WINCFG
设置底图窗口参数	NET_DVR_SetDVRConfig	命令: NET_DVR_SET_BASEMAP_WIN_CFG 结构体: NET_DVR_BASEMAP_WINCFG
获取底图图片信息	NET_DVR_GetDVRConfig	命令: NET_DVR_GET_BASEMAP_PIC_INFO 结构体: NET_DVR_BASEMAP_PIC_INFO
底图上传	NET_DVR_PicUpload	
获取上传的进度	NET_DVR_GetPicUploadProgress	
获取上传的状态	NET_DVR_GetPicUploadState	
关闭上传句柄, 释放资源	NET_DVR_CloseUploadHandle	
图片回显		
开始图片回显	NET_DVR_StartPicPreview	
停止图片回显	NET_DVR_StopScreenPic	
图像微调		
图像微调	NET_DVR_MatrixPicAdjust	
报警布防		
注册回调函数, 接收设备报警消息等	NET_DVR_SetDVRMessageCallBack_V30	支持的报警信息类型有: COMM_DVCS_STATE_ALARM
建立报警上传通道, 获取报警等信息	NET_DVR_SetupAlarmChan_V41	
撤销报警上传通道	NET_DVR_CloseAlarmChan_V30	
设备远程维护		
设置远程升级时网络环境	NET_DVR_SetNetworkEnvironment	
远程升级设备固件	NET_DVR_Upgrade	上传升级文件数据
获取远程升级的进度	NET_DVR_GetUpgradeProgress	
获取远程升级的状态	NET_DVR_GetUpgradeState	
关闭远程升级句柄, 释放资源	NET_DVR_CloseUpgradeHandle	
开始升级设备	NET_DVR_StartUpgrade	
停止升级设备	NET_DVR_StopUpgrade	
查找日志信息	NET_DVR_FindDVRLog_V30	
逐条获取查找到的日志信息	NET_DVR_FindNextLog_V30	
关闭日志查找, 释放资源	NET_DVR_FindLogClose_V30	

重启设备	NET_DVR_SetDeviceConfig	命令：NET_DVR_REBOOT_V40
恢复默认参数	NET_DVR_SetDeviceConfig	命令：NET_DVR_RESTORE_V40
导出配置文件	NET_DVR_GetConfigFile	
导入配置文件	NET_DVR_SetConfigFile	

6 函数说明

6.1 SDK 初始化

6.1.1 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](#)

6.1.2 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](#)

6.2 SDK 本地功能

SDK 本地参数配置

6.2.1 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 6.1

[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 6.1

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:

表 6.1 本地参数类型

enumType 宏定义	类型值	含义	IpOutBuff 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](#)

6.2.2 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 6.2
`[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 6.2

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 6.2 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
NET_DVR_LOCAL_CFG_TYPE_LOG	15	日志参数配置	NET_DVR_LOCAL_LOG_CFG

[返回目录](#)

连接和接收超时时间及重连设置

6.2.3 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](#)

6.2.4 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](#)

6.2.5 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](#)

多网卡绑定

6.2.6 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](#)

6.2.7 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](#)

SDK 版本、状态和能力

6.2.8 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](#)

6.2.9 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 SDK 状态信息, 详见结构体: 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](#)

6.2.10 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 SDK 功能信息, 详见结构体: 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](#)

SDK 启用写日志

6.2.11 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, 更改目录时到下一次写文件时才会使用新的目录写文件。
- bAutoDel 为 TRUE 时表示覆盖模式, 日志文件个数超过 SDK 限制个数时将会自动删除超出的文件。SDK 限制个数默认为 10 个, 可以调用接口 [NET_DVR_SetSDKLocalCfg](#)(配置类型: NET_DVR_LOCAL_CFG_TYPE_LOG)进行修改配置。

[返回目录](#)

异常消息回调

6.2.12 设置异常消息回调函数 **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 IUserID, LONG IHandle, void *pUser)
```

[out]dwType 异常或重连等消息的类型，详见表 6.3
 [out]IUserID 登录 ID
 [out]IHandle 出现异常的相应类型的句柄
 [out]pUser 用户数据

表 6.3 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_DISKFMT	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（见上表）；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_ALARM:                //报警上传时网络异常
            sprintf(tempbuf,"报警上传时网络异常!!!");
            TRACE("%s",tempbuf);
            //TODO: 关闭报警上传
            break;
        case EXCEPTION_EXCHANGE:            //用户交互时异常
            sprintf(tempbuf,"用户交互时网络异常!!!");
            TRACE("%s",tempbuf);
            //TODO: 注销登录
            break;
        default:
            break;
    }
};
```

[返回目录](#)

获取错误信息

6.2.13 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。

[返回目录](#)

6.2.14 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:

[返回目录](#)

6.3 用户注册

6.3.1 激活设备 **NET_DVR_ActivateDevice**

函 数： BOOL NET_DVR_ActivateDevice(char* sDVRIP, WORD wDVRPort, LPNET_DVR_ACTIVATECFG lpActivateCfg)

参 数： [in]sDVRIP 设备 IP 地址
[in]wDVRPort 设备端口
[in]lpActivateCfg 激活参数，包括激活使用的初始密码

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： 出厂设备需要先激活，然后再使用激活使用的初始密码登录设备。

[返回目录](#)

6.3.2 DDNS 域名和 IP Server 解析 **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
[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
[in]wDVRNameLen 设备名称的长度
[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。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_V40](#) 登录设备。
支持的解析服务器有 IPServer 和 hiDDNS。

[返回目录](#)

6.3.3 用户注册设备 **NET_DVR_Login_V40**

函 数： LONG NET_DVR_Login_V40(LPNET_DVR_USER_LOGIN_INFO pLoginInfo, LPNET_DVR_DEVICEINFO_V40 lpDeviceInfo)

参 数： [in]pLoginInfo 登录参数，包括设备地址、登录用户、密码等，详见结构体：
[NET_DVR_USER_LOGIN_INFO](#)
[out]lpDeviceInfo 设备信息(同步登录即 pLoginInfo 中 bUseAsynLogin 为 0 时有效)，详见结构体：
[NET_DVR_DEVICEINFO_V40](#)

返回值： 异步登录的状态、用户 ID 和设备信息通过 NET_DVR_USER_LOGIN_INFO 结构体中设置的回调函数(fLoginResultCallBack)返回。对于同步登录，接口返回-1 表示登录失败，其他值表示返回的用户 ID 值。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：

- pLoginInfo 中 bUseAsynLogin 为 0 时登录为同步模式，接口返回成功即表示登录成功；pLoginInfo 中 bUseAsynLogin 为 1 时登录为异步模式，登录是否成功在输入参数设置的回调函数中返回。
- 设备同时最多允许 128 个用户注册。
- SDK 支持 2048 个注册，返回 UserID 的取值范围为 0~2047。

[返回目录](#)

6.3.4 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_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](#)

6.4 设备能力集

6.4.1 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] lUserID 登录主控板，NET_DVR_Login_V40 的返回值

[in] dwAbilityType 能力类型，具体定义见表 6.4

[in] pInBuf 输入缓冲区指针，参见表 6.5

[in] dwInLength 输入缓冲区的长度

[out] pOutBuf 输出缓冲区指针，参见表 6.5

[in] dwOutLength 接收数据的缓冲区的长度

表 6.4 设备能力集类型

dwAbilityType 宏定义	宏定义值	含义
WALL_ABILITY	0x212	电视墙能力集
DECODER_ABILITY	0x261	解码器 xml 能力集
DEVICE_SOFTWARE_ABILITY	0x001	设备软硬件能力集
DEVICE_ENCODE_ALL_ABILITY_V20	0x008	设备所有编码能力集
DEVICE_USER_ABILITY	0x00c	设备用户管理参数能力集
DEVICE_NETAPP_ABILITY	0x00d	设备网络应用参数能力集
DEVICE_VIDEOPIC_ABILITY	0x00e	设备图像参数能力集

DEVICE_JPEG_CAP_ABILITY	0x00f	设备 JPEG 抓图能力集
DEVICE_SERIAL_ABILITY	0x010	设备 RS232 和 RS485 串口能力集
DEVICE_ABILITY_INFO	0x011	设备通用能力类型，具体能力根据发送的能力节点来区分
CAM_FUSION_ABILITY	0x803	相机拼接能力集
OPTICAL_DEV_ACCESS_ABILITY	0x805	光端机接入能力集
NET_RING_ABILITY	0x806	光端机环网能力集

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 6.5 所示。

表 6.5 设备能力集

能力类型宏定义	能力类型说明	pInBuf	pOutBuf
WALL_ABILITY	获取电视墙能力集	无	电视墙能力 XML 描述(WallAbility)
DECODER_ABILITY	解码器 xml 能力集	解码器能力集获取输入描述	解码器能力集 XML 描述 (DecoderAbility)
DEVICE_SOFTHARDWARE_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	获取设备所有编码能力	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)
DEVICE_VIDEOPIC_ABILITY	获取设备图像参数能力	图像参数能力获取输入描述	设备图像参数能力 XML 描述 (VideoPicAbility)
DEVICE_JPEG_CAP_ABILITY	获取设备 JPEG 抓图能力	JPEG 抓图能力获取输入描述	设备 JPEG 抓图能力 XML 描述 (JpegCaptureAbility)
DEVICE_SERIAL_ABILITY	获取设备 RS232 和 RS485 串口能力	串口能力获取输入描述	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	获取报警事件处理能力集	报警事件处理能力 XML 描述 (EventAbility)
		获取 GB/T28181 能力集	设备 GB/T28181 能力 XML 描述 (GBT28181AccessAbility)
CAM_FUSION_ABILITY	获取相机拼接能力	相机拼接能力集获取输入描述	相机拼接能力 XML 描述 (CamFusionAbility)
OPTICAL_DEV_ACCESS_ABILITY	光端机接入能力集	光端机接入能力集获取输入描述	光端机接入能力集 XML 描述 (OpticalDevAccessAbility)
NET_RING_ABILITY	获取光端机环网能力集	环网能力集获取输入描述	环网能力 XML 描述(NetRingAbility)

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](#)

6.5 布防、撤防

设置报警等信息上传的回调函数

6.5.1 注册报警信息回调函数 **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 6.6

[out] pAlarmer 报警设备信息，详见结构体 [NET_DVR_ALARMER](#)

[out] pAlarmInfo 报警信息，详见表 6.7

[out] dwBufLen 报警信息缓存大小

[out] pUser 用户数据

表 6.6 报警布防信息类型

ICommand 宏定义	宏定义值	含义
COMM_ALARM_VIDEOPLATFORM	0x1104	视频综合平台报警
COMM_SCREEN_ALARM	0x5000	多屏控制器报警信息上传
COMM_DVCS_STATE_ALARM	0x5001	分布式多屏控制器报警上传
COMM_ALARM_WALL_CONFERNECE	0x5015	MCU 单个已开会的会议信息报警

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 6.7。

表 6.7 报警信息结构

消息类型（ICommand）	上传内容	pAlarmInfo 对应的结构体
COMM_ALARM_VIDEOPLATFORM	视频综合平台报警	NET_DVR_VIDEOPLATFORM_ALRAMINFO
COMM_SCREEN_ALARM	多屏控制器上传的报警信息	NET_DVR_SCREENALARMCFG
COMM_DVCS_STATE_ALARM	分布式多屏控制器上传报警信息	NET_DVR_DVCS_STATE_ALARM
COMM_ALARM_WALL_CONFERNECE	MCU 单个已开会的会议信息报警	WallConference

[返回目录](#)

布防撤防

6.5.2 报警布防，建立报警上传通道 **NET_DVR_SetupAlarmChan_V41**

函 数: LONG NET_DVR_SetupAlarmChan_V41(LONG IUserID, LPNET_DVR_SETUPALARM_PARAM lpSetupParam)

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V40 的返回值
[in] IpSetupParam 报警布防参数, 详见结构体: [NET_DVR_SETUPALARM_PARAM](#), 其中结构体大小需要赋值, 其他参数不支持 (设为 0)

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: 使用该接口支持上传 V3.0 以上版本支持的设备的报警结构。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等信息。

[返回目录](#)

6.5.3 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](#)

6.6 监听报警

6.6.1 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 6.8
[out] pAlarmer 报警设备信息
[out] pAlarmInfo 报警信息, 详见表 6.9
[out] dwBufLen 报警信息缓存大小
[out] pUser 用户数据

表 6.8 报警监听消息类型

ICommand 宏定义	宏定义值	含义
COMM_ALARM_VIDEOPLATFORM	0x1104	视频综合平台报警

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 6.9。

表 6.9 报警信息结构

消息类型 (ICommand)	上传内容	pAlarmInfo 对应的结构体
COMM_ALARM_VIDEOPLATFORM	视频综合平台报警	NET_DVR_VIDEOPLATFORM_ALARMINFO

[返回目录](#)

6.6.2 停止监听（支持多线程）**NET_DVR_StopListen_V30**

函 数： BOOL NET_DVR_StopListen_V30(LONG IListenHandle)

参 数： [in] IListenHandle 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](#)

6.7 通用参数配置

6.7.1 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_V40 的返回值

[in] dwCommand 设备配置命令，详见表 6.10

[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 6.10

[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 6.10 所示。

表 6.10 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_ZONEANDDST	获取时区和夏时制参数	无效	NET_DVR_ZONEANDDST	128
NET_DVR_GET_NETCFG_V30	获取网络参数	无效	NET_DVR_NETCFG_V30	1000
NET_DVR_GET_USERCFG_V40	获取用户参数	组号，从 0	NET_DVR_USER_V40	6187

		开始, 每组 32 个用户		
NET_DVR_GET_EXCEPTIONCFG_V30	获取异常参数	无效	NET_DVR_EXCEPTION_V30	1034
NET_DVR_VIDEOPLATFORMALARMCFG_GET	获取视频综合平台报警配置	无效	NET_DVR_VIDEOPLATFORMALARMCFG	1130
NET_DVR_GET_ETHERNET_CHANNEL	获取端口聚合参数	无效	NET_DVR_ETHERNET_CHANNEL_INFO	1306
NET_DVR_BIGSCREEN_GETSERIAL	获取大屏串口信息	无效	NET_DVR_SERIAL_CONTROL	1614
NET_DVR_GET_VCS_USER_CFG	获取大屏用户信息	用户号, 从 1 开始	NET_DVR_VCS_USER_INFO	1623
NET_DVR_GET_USING_SERIALPORT	获取当前使用串口	无效	NET_DVR_USING_SERIALPORT	1742
NET_DVR_GET_MAINBOARD_SERIAL	获取主控板串口参数	串口号	NET_DVR_MAINBOARD_SERIAL_CFG	1757
NET_DVR_GET_CAMERACHAN_SERIALCFG	获取通道绑定串口配置	编码通道号 (组合)	NET_DVR_CAMERACHAN_SERIALCFG	1760
NET_DVR_GET_GBT28181_ACCESS_CFG	获取 GBT28181 协议接入配置	无效	NET_DVR_GBT28181_ACCESS_CFG	3249
NET_DVR_GET_WIN_ROAM_SWITCH_CFG	获取解码器窗口漫游开关参数	无效	NET_DVR_WIN_ROAM_SWITCH_CFG	9224

[返回目录](#)

6.7.2 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_V40 的返回值
[in] dwCommand 设备配置命令, 详见表 6.11
[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 6.11
[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 6.11 所示。

表 6.11 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	设置时间参数	无效	NET_DVR_TIME	119
NET_DVR_SET_ZONEANDDST	设置时区和夏时制参数	无效	NET_DVR_ZONEANDDST	129
NET_DVR_SET_NETCFG_V30	设置网络参数	无效	NET_DVR_NETCFG_V30	1001
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_EXCEPTIONCFG_V30	设置异常参数	无效	NET_DVR_EXCEPTION_V30	1035

NET_DVR_VIDEOPLATFARMALARMCFG_SET	设置视频综合平台报警配置	无效	NET_DVR_VIDEOPLATFARMALARMCFG	1131
NET_DVR_SET_ETHERNET_CHANNEL	设置端口聚合参数	无效	NET_DVR_ETHERNET_CHANNEL_INFO	1307
NET_DVR_SET_VCS_USER_CFG	设置大屏用户信息	用户号, 从 1 开始	NET_DVR_VCS_USER_INFO	1624
NET_DVR_SET_USING_SERIALPORT	设置当前使用串口	无效	NET_DVR_USING_SERIALPORT	1743
NET_DVR_SET_MAINBOARD_SERIAL	设置主控板串口参数	串口号	NET_DVR_MAINBOARD_SERIAL_CFG	1756
NET_DVR_SET_CAMERACHAN_SERIALCFG	设置通道绑定串口配置	编码通道号 (组合)	NET_DVR_CAMERACHAN_SERIALCFG	1761
NET_DVR_SET_GBT28181_ACCESS_CFG	设置 GBT28181 协议接入配置	无效	NET_DVR_GBT28181_ACCESS_CFG	3250
NET_DVR_SET_WIN_ROAM_SWITCH_CFG	设置解码器窗口漫游开关参数	无效	NET_DVR_WIN_ROAM_SWITCH_CFG	9225

[返回目录](#)

6.7.3 批量获取设备参数 **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 登录主控板, NET_DVR_Login_V40 的返回值
 - [in] dwCommand 设备配置命令, 详见表 6.12
 - [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 6.13
 - [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 6.13), 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 6.12 批量参数获取命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_GET_GBT28181_CHANINFO_CFG	获取 GBT28181 协议接入设备的通道信息	3251
NET_DVR_GET_GBT28181_DECCHANINFO_CFG	获取 GBT28181 协议接入设备的解码通道信息	1754

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监控点配置信息的通用接口。lpInBuffer 指定需要获取的 dwCount 个配置信息, lpOutBuffer 保存获取到的 dwCount 个配置信息。
全部获取时 dwCount 置为 0xffffffff, lpInBuffer 置为 NULL, dwInBufferSize 置为 0, lpStatusList 置为 NULL, lpOutBuffer 前 4 个字节为返回的个数。如果设置的 lpOutBuffer 缓冲区不足, 仅返回部分信息, 可以根据返回的个数 (前 4 字节的值) 重新获取。

表 6.13 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_GBT28181_CHANINFO_CFG	dwCount 个 NET_DVR_STREAM_INFO	dwCount 个 NET_DVR_GBT28181_CHANINFO_CFG
NET_DVR_GET_GBT28181_DECCHANINFO_CFG	dwCount 个 NET_DVR_STREAM_INFO	dwCount 个 NET_DVR_GBT28181_CHANINFO_CFG

[返回目录](#)

6.7.4 批量设置设备参数 **NET_DVR_SetDeviceConfig**

- 函 数: BOOL NET_DVR_SetDeviceConfig(LONG lUserID,DWORD dwCommand,DWORD dwCount,LPVOID lpInBuffer,DWORD dwInBufferSize,LPVOID lpStatusList, LPVOID lpInParamBuffer,DWORD dwInParamBufferSize)
- 参 数:
- [in] lUserID 登录主控板, NET_DVR_Login_V40 的返回值
 - [in] dwCommand 设备配置命令, 详见表 6.14
 - [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] lpInBuffer 配置条件缓冲区, 详见表 6.15
 - [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
 - [in] lpInParamBuffer 需要设置给设备的参数内容 (详见表 6.15), 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, 表示对应的 lpInBuffer 设置失败, 为 0 则设置成功
 - [in] dwInParamBufferSize 设置内容缓冲区大小

表 6.14 批量参数配置命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_SET_GBT28181_CHANINFO_CFG	设置 GBT28181 协议接入设备的通道信息	3252
NET_DVR_SET_GBT28181_DECCHANINFO_CFG	设置 GBT28181 协议接入设备的解码通道信息	1755

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。lpInBuffer 指定需要设置的 dwCount 个监控点信息, lpOutBuffer 保存将要设置的 dwCount 个监控点的配置信息。不同的获取功能对应不同的结构体和命令号, 如表 6.15 所示。

表 6.15 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_GBT28181_CHANINFO_CFG	dwCount 个 NET_DVR_STREAM_INFO	dwCount 个 NET_DVR_GBT28181_CHANINFO_CFG
NET_DVR_SET_GBT28181_DECCHANINFO_CFG	dwCount 个 NET_DVR_STREAM_INFO	dwCount 个 NET_DVR_GBT28181_CHANINFO_CFG

[返回目录](#)

6.8 信号源和设备管理

视频综合平台子系统信息

6.8.1 获取所有子系统信息 **NET_DVR_MatrixGetSubSystemInfo_V40**

函 数： BOOL NET_DVR_MatrixGetSubSystemInfo_V40(LONG IUserID, LPNET_DVR_ALLSUBSYSTEMINFO_V40 lpInter)

参 数： [in] IUserID 登录主控板，NET_DVR_Login_V40 的返回值
[out] lpInter 子系统信息，详见结构体 [NET_DVR_ALLSUBSYSTEMINFO_V40](#)

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： 视频综合平台支持。

[返回目录](#)

6.8.2 设置子系统参数 **NET_DVR_MatrixSetSubSystemInfo_V40**

函 数： BOOL NET_DVR_MatrixSetSubSystemInfo_V40 (LONG IUserID, LPNET_DVR_ALLSUBSYSTEMINFO_V40 lpInter)

参 数： [in] IUserID 登录主控板，NET_DVR_Login_V40 的返回值
[in] lpInter 子系统信息，详见结构体 [NET_DVR_ALLSUBSYSTEMINFO_V40](#)

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： 视频综合平台支持。

[返回目录](#)

信号源管理

6.8.3 获取信号源列表 **NET_DVR_GetInputSignalList_V40**

函 数： BOOL NET_DVR_GetInputSignalList_V40(LONG IUserID, DWORD dwDevNum, LPNET_DVR_INPUT_SIGNAL_LIST lpInputSignalList)

参 数： [in] IUserID 登录主控板，NET_DVR_Login_V40 返回值
[in] dwDevNum 保留，设为 0
[out] lpInputSignalList 信号源列表，详见结构体：[NET_DVR_INPUT_SIGNAL_LIST](#)

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： 1.调用该接口，将 NET_DVR_INPUT_SIGNAL_LIST 中的 pBuffer 置为空且 dwBufLen 为 0，此时获取输入信号源个数（dwInputSignalNums）；
2.根据第一次调用接口返回的数量分配 pBuffer 的大小，然后再次调用该接口，此时获取输入信号源列表（pBuffer）。

[返回目录](#)

6.8.4 设置信号源参数 **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_V40 的返回值
[in] dwCommand 设备配置命令, 详见表 6.16
[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 6.16
[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 6.16 所示。

表 6.16 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_INPUTSTREAMCFG_V40	设置输入信号源	4 字节表示信号源序号	NET_DVR_INPUTSTREAMCFG_V40	1776

[返回目录](#)

6.8.5 批量获取信号源参数 **NET_DVR_GetDeviceConfig**

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

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V40 的返回值
[in] dwCommand 设备配置命令, 详见表 6.17
[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 6.18
[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 6.18), 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 6.17 批量参数获取命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_GET_EXTERNAL_MATRIX_CFG	获取矩阵信息	1619
NET_DVR_GET_DEV_NETINFO	获取解码板网络参数	1652
NET_DVR_GET_SIGNAL_SOURCE_INFO	获取信号源信息	1654

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

因。

说明： 该接口是带有发送数据的批量获取监控点配置信息的通用接口。IpInBuffer 指定需要获取的 dwCount 个配置信息，IpOutBuffer 保存获取到的 dwCount 个配置信息。
全部获取时 dwCount 置为 0xffffffff，IpInBuffer 置为 NULL，dwInBufferSize 置为 0，IpStatusList 置为 NULL，IpOutBuffer 前 4 个字节为返回的个数。如果设置的 IpOutBuffer 缓冲区不足，仅返回部分信息，可以根据返回的个数（前 4 字节的值）重新获取。

表 6.18 批量参数获取结构

dwCommand 宏定义	IpInBuffer 对应结构体	IpOutBuffer 对应结构体
NET_DVR_GET_EXTERNAL_MATRIX_CFG	dwCount 个矩阵 ID，每个 ID 为 4 个字节	dwCount 个 NET_DVR_EXTERNAL_MATRIX_CFG
NET_DVR_GET_DEV_NETINFO	dwCount 个 NET_DVR_DEVICEID_INFO	dwCount 个 NET_DVR_NETPARAM
NET_DVR_GET_SIGNAL_SOURCE_INFO	dwCount 个 NET_DVR_DEVICEID_INFO	dwCount 个 NET_DVR_CHAN_INFO

[返回目录](#)

6.8.6 批量设置信号源参数 **NET_DVR_SetDeviceConfig**

函数： BOOL NET_DVR_SetDeviceConfig(LONG lUserID,DWORD dwCommand,DWORD dwCount,LPVOID IpInBuffer,DWORD dwInBufferSize,LPVOID IpStatusList, LPVOID IpInParamBuffer,DWORD dwInParamBufferSize)

参数：

[in] lUserID	登录主控板，NET_DVR_Login_V40 的返回值
[in] dwCommand	设备配置命令，详见表 6.19
[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] IpInBuffer	配置条件缓冲区，详见表 6.20
[in] dwInBufferSize	缓冲区长度
[out] IpStatusList	错误信息列表，和要查询的监控点一一对应，例如 IpStatusList[2] 就对应 IpInBuffer[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
[in] IpInParamBuffer	需要设置给设备的参数内容（详见表 6.20），和要查询的监控点一一对应。如果某个监控点对应的 IpStatusList 信息为大于 0 值，表示对应的 IpInBuffer 设置失败，为 0 则设置成功
[in] dwInParamBufferSize	设置内容缓冲区大小

表 6.19 批量参数配置命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_SET_EXTERNAL_MATRIX_CFG	获取矩阵信息	1620
NET_DVR_SET_DEV_NETINFO	设置解码板网络参数	1653
NET_DVR_SET_SIGNAL_SOURCE_INFO	设置信号源信息	1655

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 IpStatusList[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。IpInBuffer 指定需要设置的 dwCount 个监控点信息，IpOutBuffer 保存将要设置的 dwCount 个监控点的配置信息。不同的获取功能对应不同的结构体和命令号，如表 6.20 所示。

表 6.20 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_EXTERNAL_MATRIX_CFG	dwCount 个矩阵 ID，每个 ID 为 4 个字节	dwCount 个 NET_DVR_EXTERNAL_MATRIX_CFG
NET_DVR_SET_DEV_NETINFO	dwCount 个 NET_DVR_DEVICEID_INFO	dwCount 个 NET_DVR_NETPARAM
NET_DVR_SET_SIGNAL_SOURCE_INFO	dwCount 个 NET_DVR_DEVICEID_INFO	dwCount 个 NET_DVR_CHAN_INFO

[返回目录](#)

子板和子板异常信息

6.8.7 获取子板和子板异常信息 **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_V40 的返回值
[in] dwCommand 设备配置命令，详见表 6.21
[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 6.21
[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 6.21 所示。

- 获取子板信息后，对异常的子板，可以通过 NET_DVR_GET_SUBBOARD_EXCEPTION 查看子板异常具体信息。
- 获取子板异常信息时接口所需的子板槽位号(IChannel)，就是 NET_DVR_GET_SUBBOARD_INFO 获取子板信息中的子板槽位号(dwSlotNo)。

表 6.21 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_SUBBOARD_INFO	获取子板信息	无效	NET_DVR_DEVICE_SUBBOARD_INFO	1758
NET_DVR_GET_SUBBOARD_EXCEPTION	获取异常子板异常信息	子板槽位号(组合)	NET_DVR_DEVICE_BOARD_EXCEPINFO	1759

[返回目录](#)

分布式多屏控制器子设备

6.8.8 获取分布式子设备信息列表 **NET_DVR_GetDevList**

函 数: BOOL NET_DVR_GetDevList(LONG IUserID, LPNET_DVR_DEVLIST lpDevListInfo)

参 数: [in] IUserID 登录主控板，NET_DVR_Login_V40 返回值
[out] lpDevListInfo 分布式子设备信息列表，详见结构体: [NET_DVR_DEVLIST](#)

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：**
- 将 NET_DVR_DEVLIST 中的 pBuffer 置为空且 dwBufLen 为 0，此时只返回分布式子设备数量。然后上层客户端根据返回的数量分配 pBuffer 的大小，然后再次调用该接口，获取设备列表。
 - 该接口获取所有子设备列表，而通过接口 [NET_DVR_GetDeviceConfig](#)（命令：NET_DVR_GET_DEV_BASEINFO）可以获取单个子设备的信息。
 - 设置子设备通过接口 [NET_DVR_SetDeviceConfig](#)（命令：NET_DVR_SET_DEV_BASEINFO）实现。

[返回目录](#)

6.8.9 批量获取子设备信息 NET_DVR_GetDeviceConfig

函数： BOOL NET_DVR_GetDeviceConfig(LONG IUserID, DWORD dwCommand, DWORD dwCount, LPVOID lpInBuffer, DWORD dwInBufferSize, LPVOID lpStatusList, LPVOID lpOutBuffer, DWORD dwOutBufferSize)

参数：

[in] IUserID	登录主控板，NET_DVR_Login_V40 的返回值
[in] dwCommand	设备配置命令，详见表 6.22
[in] dwCount	要设置的子设备个数
[in] lpInBuffer	配置条件缓冲区，详见表 6.23
[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 6.23），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 6.22 批量获取参数命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_GET_DEV_BASEINFO	获取单个设备信息	1650

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 个监控点信息，lpOutBuffer 保存获取到的 dwCount 个监控点的配置信息。不同的获取功能对应不同的结构体和命令号，如表 6.23 所示。

表 6.23 批量参数获取结构

dwCommand	lpInBuffer 对应结构体	lpOutBuffer 对应结构体
NET_DVR_GET_DEV_BASEINFO	dwCount 个 NET_DVR_DEVICEID_INFO	dwCount 个 NET_DVR_DEV_BASE_INFO

- dwCount 为 0xffffffff 时，lpInBuffer 为 NULL，dwInBufferSize 为 0，表示获取所有的相关信息，并且此时 lpOutBuffer 的前 4 个字节为返回的实际信息个数，4 字节后才是返回的具体信息，因此，lpOutBuffer 缓冲区的大小需要比列表所述大小多 4 个字节。

[返回目录](#)

6.8.10 批量设置子设备信息 **NET_DVR_SetDeviceConfig**

- 函 数: BOOL NET_DVR_SetDeviceConfig(LONG IUserID,DWORD dwCommand,DWORD dwCount,LPVOID lpInBuffer,DWORD dwInBufferSize,LPVOID lpStatusList, LPVOID lpInParamBuffer,DWORD dwInParamBufferSize)
- 参 数:
- [in] IUserID 登录主控板, NET_DVR_Login_V40 的返回值
 - [in] dwCommand 设备配置命令, 详见表 6.24
 - [in] dwCount 要设置的子设备个数
 - [in] lpInBuffer 配置条件缓冲区, 详见表 6.25
 - [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
 - [in] lpInParamBuffer 需要设置给设备的参数内容 (详见表 6.25), 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, 表示对应的 lpInBuffer 设置失败, 为 0 则设置成功
 - [in] dwInParamBufferSize 设置内容缓冲区大小

表 6.24 批量设置参数命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_SET_DEV_BASEINFO	设置单个设备信息	1651

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配置信息的通用接口。lpInBuffer 指定需要设置的 dwCount 个监控点信息, lpOutBuffer 保存将要设置的 dwCount 个监控点的配置信息。不同的设置功能, lpInBuffer 和 lpOutBuffer 对应不同的结构体, 如表 6.25 所示。

表 6.25 批量参数设置结构

dwCommand	lpInBuffer 对应结构体	lpInParamBuffer 对应结构体
NET_DVR_SET_DEV_BASEINFO	dwCount 个 NET_DVR_DEVICEID_INFO	dwCount 个 NET_DVR_DEV_BASE_INFO

[返回目录](#)

屏幕服务器登录参数配置

6.8.11 屏幕服务器登录参数配置 **NET_DVR_STDXMLConfig**

- 函 数: BOOL NET_DVR_STDXMLConfig(LONG IUserID, NET_DVR_XML_CONFIG_INPUT *lpInputParam, NET_DVR_XML_CONFIG_OUTPUT *lpOutputParam)
- 参 数:
- [in] IUserID 登录主控板, NET_DVR_Login_V40 返回值
 - [in] lpInputParam 输入参数
 - [out] lpOutputParam 输出参数
- 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：**
- 通过该接口可以直接透传 ISAPI 协议命令，实现参数配置、能力集等功能。调用该接口需要设备支持 ISAPI 协议（PUT、GET、POST、DELETE 等命令），不同功能对应不同的输入输出参数，具体内容可以参考 ISAPI 协议文档。
 - 屏幕服务器登录参数配置相关的输入输出参数如表 6.26 所示。
 - <ID>表示服务器 ID，一台服务器只有一个用户名和密码，不支持多用户。

表 6.26 屏幕服务器登录参数配置

功能	IpInputParam->		IpOutputParam->
	IpRequestUrl	IpInBuffer	IpOutBuffer
获取所有	GET /ISAPI/DisplayDev/Auxiliary/ScreenServer/loginCfg	NULL	ServerLoginCfgList
获取	GET /ISAPI/DisplayDev/Auxiliary/ScreenServer/<ID>/loginCfg	NULL	ServerLoginCfg
设置	PUT /ISAPI/DisplayDev/Auxiliary/ScreenServer/<ID>/loginCfg	ServerLoginCfg	NULL
添加	POST /ISAPI/DisplayDev/Auxiliary/ScreenServer/loginCfg	ServerLoginCfg	NULL
删除	DELETE /ISAPI/DisplayDev/Auxiliary/ScreenServer/<ID>/loginCfg	NULL	NULL
能力集	GET /ISAPI/DisplayDev/Auxiliary/ScreenServer/<ID>/loginCfg/capabilities	NULL	ServerLoginCfg

[返回目录](#)

输入板参数配置

6.8.12 获取输入板参数 **NET_DVR_GetSTDConfig**

函数： BOOL NET_DVR_GetSTDConfig(LONG lUserID, DWORD dwCommand, LPNET_DVR_STD_CONFIG lpConfigParam)

参数：

- [in]lUserID 用户 ID 号，NET_DVR_Login_V40 的返回值
- [in]dwCommand 设备配置命令，详见表 6.27
- [in&out]lpConfigParam 配置输入输出参数，详见结构体：[NET_DVR_STD_CONFIG](#)，不同的配置功能对应不同的输入输出参数，详见表 6.28

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： 获取配置参数时，lpConfigParam 结构体中的 lpInBuffer 无效，设为 NULL。对于不同的配置功能（dwCommand），lpConfigParam 中 lpCondBuffer、lpOutBuffer 分别对应不同的内容，具体如表 6.28 所示。

表 6.27 获取设备参数命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_GET_INPUT_BOARD_CFG	获取输入板配置信息	9281
NET_DVR_GET_INPUT_BOARD_CFG_LIST	获取输入板配置信息列表	9282

表 6.28 输入板参数

dwCommand 宏定义	lpCondBuffer	lpOutBuffer	lpXmlBuffer
NET_DVR_GET_INPUT_BOARD_CFG	输入板槽位号(组合)	NET_DVR_INPUT_BOARD_CFG	InputBoardCfg
NET_DVR_GET_INPUT_BOARD_CFG_LIST	NULL	NET_DVR_INPUT_BOARD_CFG_LIST	InputBoardCfgList

[返回目录](#)

6.8.13 设置输入板参数 **NET_DVR_SetSTDConfig**

- 函 数: BOOL NET_DVR_SetSTDConfig(LONG IUserID, DWORD dwCommand, LPNET_DVR_STD_CONFIG lpConfigParam)
- 参 数: [in]IUserID 用户 ID 号, NET_DVR_Login_V40 的返回值
[in]dwCommand 设备配置命令, 详见表 6.29
[in&out]lpConfigParam 配置输入输出参数, 详见结构体: [NET_DVR_STD_CONFIG](#), 不同的配置功能对应不同的输入输出参数, 详见表 6.30
- 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: 设置配置参数时, lpConfigParam 结构体里面的 lpOutBuffer 无效, 设为 NULL。对于不同的配置功能 (dwCommand), lpConfigParam 中的 lpCondBuffer、lpInBuffer 分别对应不同的内容, 具体如表 6.30 所示。

表 6.29 设置设备命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_SET_INPUT_BOARD_CFG	设置输入板配置信息	9283

表 6.30 输入板参数

dwCommand 宏定义	lpCondBuffer	lpInBuffer	lpXmlBuffer
NET_DVR_SET_INPUT_BOARD_CFG	输入板槽位号(组合)	NET_DVR_INPUT_BOARD_CFG	InputBoardCfg

[返回目录](#)

输入口显示参数配置

6.8.14 获取输入口显示参数能力集 **NET_DVR_GetSTDAbility**

- 函 数: BOOL NET_DVR_GetSTDAbility(LONG IUserID, DWORD dwAbilityType, LPNET_DVR_STD_ABILITY lpAbilityParam)
- 参 数: [in]IUserID 用户 ID 号, NET_DVR_Login_V40 的返回值
[in]dwAbilityType 能力类型, 具体定义见表 6.31
[in&out]lpAbilityParam 设备能力集参数 (包括输入和输出参数), 不同的能力集对应不同的输入输出参数
- 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 6.31 获取设备能力集

宏定义	宏定义值	含义	lpCondBuffer	lpOutBuffer
NET_DVR_GET_DISPINPUT_CFG_CAP	9299	获取输入口显示参数能力集	NULL	InputConfigList

[返回目录](#)

6.8.15 获取输入口显示参数 **NET_DVR_GetSTDConfig**

- 函 数：** BOOL NET_DVR_GetSTDConfig(LONG IUserID, DWORD dwCommand, LPNET_DVR_STD_CONFIG lpConfigParam)
- 参 数：** [in]IUserID 用户 ID 号，NET_DVR_Login_V40 的返回值
[in]dwCommand 设备配置命令，详见表 6.32
[in&out]lpConfigParam 配置输入输出参数，详见结构体：[NET_DVR_STD_CONFIG](#)，不同的配置功能对应不同的输入输出参数，详见表 6.32
- 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：** 获取配置参数时，lpConfigParam 结构体中的 lpInBuffer 无效，设为 NULL。对于不同的配置功能（dwCommand），lpConfigParam 中 lpCondBuffer、lpOutBuffer 分别对应不同的内容，具体如表 6.32 所示。

表 6.32 获取设备参数

dwCommand 宏定义	dwCommand 含义	lpCondBuffer	lpOutBuffer	宏定义值
NET_DVR_GET_DISPINPUT_CFG	获取输入口显示参数	4 字节输入口号(组合)	NET_DVR_DISPINPUT_CFG	9296
NET_DVR_GET_DISPINPUT_CFG_LIST	获取所有输入口显示参数	NULL	NET_DVR_DISPINPUT_CFG_LIST	9297

[返回目录](#)

6.8.16 设置输入口显示参数 **NET_DVR_SetSTDConfig**

- 函 数：** BOOL NET_DVR_SetSTDConfig(LONG IUserID, DWORD dwCommand, LPNET_DVR_STD_CONFIG lpConfigParam)
- 参 数：** [in]IUserID 用户 ID 号，NET_DVR_Login_V40 的返回值
[in]dwCommand 设备配置命令，详见表 6.33
[in&out]lpConfigParam 配置输入输出参数，详见结构体：[NET_DVR_STD_CONFIG](#)，不同的配置功能对应不同的输入输出参数，详见表 6.33
- 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：** 设置配置参数时，lpConfigParam 结构体里面的 lpOutBuffer 无效，设为 NULL。对于不同的配置功能（dwCommand），lpConfigParam 中的 lpCondBuffer、lpInBuffer 分别对应不同的内容，具体如表 6.33 所示。

表 6.33 获取设备参数

dwCommand 宏定义	dwCommand 含义	lpCondBuffer	lpInBuffer	宏定义值
NET_DVR_SET_DISPINPUT_CFG	设置输入口显示参数	4 字节输入口号(组合)	NET_DVR_DISPINPUT_CFG	9298

[返回目录](#)

虚拟屏配置

6.8.17 获取虚拟屏参数 **NET_DVR_GetSTDConfig**

函 数： BOOL NET_DVR_GetSTDConfig(LONG IUserID, DWORD dwCommand, LPNET_DVR_STD_CONFIG lpConfigParam)

参 数： [in]IUserID 用户 ID 号，NET_DVR_Login_V40 的返回值
[in]dwCommand 设备配置命令，详见表 6.34
[in&out]lpConfigParam 配置输入输出参数，详见结构体：[NET_DVR_STD_CONFIG](#)，不同的配置功能对应不同的输入输出参数，详见表 6.34

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： 获取配置参数时，lpConfigParam 结构体中的 lpInBuffer 无效，设为 NULL。对于不同的配置功能（dwCommand），lpConfigParam 中 lpCondBuffer、lpOutBuffer 分别对应不同的内容，具体如表 6.34 所示。

表 6.34 获取设备参数

dwCommand 宏定义	dwCommand 含义	lpCondBuffer	lpOutBuffer	宏定义值
NET_DVR_GET_VS_INPUT_CHAN_INIT_ALL	获取虚拟屏子板的所有输入通道的初始化参数	4 字节虚拟屏子板号	NET_DVR_VS_INPUT_CHAN_INIT_LIST	9133
NET_DVR_GET_VS_INPUT_CHAN_INIT	获取虚拟屏输入通道的初始化参数	4 字节输入口号(组合)	NET_DVR_VS_INPUT_CHAN_INIT	9134
NET_DVR_GET_VS_INPUT_CHAN_CFG	获取虚拟屏输入通道配置参数	4 字节输入口号(组合)	NET_DVR_VS_INPUT_CHAN_CFG	9135
NET_DVR_GET_VS_NETSRC_CFG	获取虚拟屏网络源参数	4 字节输入口号(组合)	NET_DVR_VS_NETSRC_CFG	9158

[返回目录](#)

6.8.18 设置虚拟屏参数 **NET_DVR_SetSTDConfig**

函 数： BOOL NET_DVR_SetSTDConfig(LONG IUserID, DWORD dwCommand, LPNET_DVR_STD_CONFIG lpConfigParam)

参 数： [in]IUserID 用户 ID 号，NET_DVR_Login_V40 的返回值
[in]dwCommand 设备配置命令，详见表 6.35
[in&out]lpConfigParam 配置输入输出参数，详见结构体：[NET_DVR_STD_CONFIG](#)，不同的配置功能对应不同的输入输出参数，详见表 6.35

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： 设置配置参数时，lpConfigParam 结构体里面的 lpOutBuffer 无效，设为 NULL。对于不同的配置功能（dwCommand），lpConfigParam 中的 lpCondBuffer、lpInBuffer 分别对应不同的内容，具体如表 6.35 所示。

表 6.35 设置设备参数

dwCommand 宏定义	dwCommand 含义	IpCondBuffer	IpInBuffer	宏定义值
NET_DVR_SET_VS_INPUT_CHAN_INIT_ALL	设置虚拟屏子板的所有输入通道的初始化参数	4 字节虚拟屏子板号	NET_DVR_VS_INPUT_CHAN_INIT_LIST	9132
NET_DVR_SET_VS_INPUT_CHAN_CFG	设置虚拟屏输入通道配置参数	4 字节输入口号(组合)	NET_DVR_VS_INPUT_CHAN_CFG	9157
NET_DVR_SET_VS_NETSRC_CFG	设置虚拟屏网络源参数	4 字节输入口号(组合)	NET_DVR_VS_NETSRC_CFG	9159

[返回目录](#)

6.8.19 获取虚拟屏参数 NET_DVR_GetDVRConfig

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

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V40 的返回值
[in] dwCommand 设备配置命令, 详见表 6.36
[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] IpOutBuffer 接收数据的缓冲指针, 详见表 6.36
[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 6.36 所示。

表 6.36 参数获取命令列表

dwCommand 宏定义	dwCommand 含义	IChannel	IpOutBuffer 对应结构体	宏定义值
NET_DVR_GET_VIRTUAL_SCREEN_CFG	获取超高清输入子系统(虚拟屏)参数	输入口号(组合)	NET_DVR_VIRTUAL_SCREEN_CFG	9122

[返回目录](#)

6.8.20 设置虚拟屏参数 NET_DVR_SetDVRConfig

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

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V40 的返回值
[in] dwCommand 设备配置命令, 详见表 6.37
[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] IpInBuffer 输入数据的缓冲指针, 详见表 6.37
[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 6.37 所示。

表 6.37 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_VIRTUAL_SCREEN_CFG	设置超高清输入子系统(虚拟屏)参数	输入口号(组合)	NET_DVR_VIRTUAL_SCREEN_CFG	9123

[返回目录](#)

6.9 电视墙操作

6.9.1 获取电视墙相关参数 **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_V40 的返回值
[in] dwCommand 设备配置命令, 详见表 6.38
[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 6.38
[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 6.38 所示。

表 6.38 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_WALL_CFG	获取电视墙全局参数	无效	NET_DVR_WALL_CFG	9014
NET_DVR_GET_VIDEOWALLDISPLAYMODE	获取电视墙拼接模式	电视墙号(组合)	NET_DVR_VIDEOWALLDISPLAYMODE	1731
NET_DVR_GET_VIDEOWALLDISPLAYNO	获取设备显示输出号	无效	NET_DVR_DISPLAYCFG	1732
NET_DVR_GET_VIRTUALLED	获取虚拟 LED 信息	虚拟 LED 号(组合)	NET_DVR_VIRTUALLED_PARAM	1739

[返回目录](#)

6.9.2 设置电视墙相关参数 **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_V40 的返回值
[in] dwCommand 设备配置命令, 详见表 6.39
[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 6.39
[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 6.39 所示。

表 6.39 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_WALL_CFG	设置电视墙全局参数	无效	NET_DVR_WALL_CFG	9015
NET_DVR_SET_VIDEOWALLDISPLAYMODE	设置电视墙拼接模式	电视墙号(组合)	NET_DVR_VIDEOWALLDISPLAYMODE	1730
NET_DVR_SET_VIRTUALLED	设置虚拟 LED 参数	虚拟 LED 号(组合)	NET_DVR_VIRTUALLED_PARAM	1738

[返回目录](#)

6.9.3 批量获取显示输出信息 **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 登录主控板，NET_DVR_Login_V40 的返回值
- [in] dwCommand 设备配置命令，详见表 6.40
- [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 6.41
- [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 6.41），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 6.40 批量参数获取命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_GET_VIDEOWALLDISPLAYPOSITION	获取显示输出位置参数	1734
NET_DVR_WALLOUTPUT_GET	获取电视墙显示输出参数	9011

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监控点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。
全部获取时 dwCount 置为 0xffffffff，lpInBuffer 置为 NULL，dwInBufferSize 置为 0，lpStatusList 置为 NULL，lpOutBuffer 前 4 个字节为返回的个数。如果设置的 lpOutBuffer 缓冲区不足，仅返回部分信息，可以根据返回的个数（前 4 字节的值）重新获取。

表 6.41 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_VIDEOWALLDISPLAYPOSITION	dwCount 个显示输出号(组合)	dwCount 个 NET_DVR_VIDEOWALLDISPLAYPOSITION
NET_DVR_WALLOUTPUT_GET	dwCount 个显示输出号(组合)	dwCount 个 NET_DVR_WALLOUTPUTPARAM

[返回目录](#)

6.9.4 批量设置显示输出参数 **NET_DVR_SetDeviceConfig**

- 函 数: BOOL NET_DVR_SetDeviceConfig(LONG IUserID,DWORD dwCommand,DWORD dwCount,LPVOID lpInBuffer,DWORD dwInBufferSize,LPVOID lpStatusList, LPVOID lpInParamBuffer,DWORD dwInParamBufferSize)
- 参 数:
- [in] IUserID 登录主控板，NET_DVR_Login_V40 的返回值
 - [in] dwCommand 设备配置命令，详见表 6.42
 - [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] lpInBuffer 配置条件缓冲区，详见表 6.43
 - [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
 - [in] lpInParamBuffer 需要设置给设备的参数内容（详见表 6.43），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，表示对应的 lpInBuffer 设置失败，为 0 则设置成功
 - [in] dwInParamBufferSize 设置内容缓冲区大小

表 6.42 批量参数配置命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_SET_VIDEOWALLDISPLAYPOSITION	设置显示输出位置参数	1733
NET_DVR_WALLOUTPUT_SET	设置电视墙显示输出参数	9012

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。lpInBuffer 指定需要设置的 dwCount 个监控点信息，lpOutBuffer 保存将要设置的 dwCount 个监控点的配置信息。

表 6.43 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_VIDEOWALLDISPLAYPOSITION	dwCount 个显示输出号(组合)	dwCount 个 NET_DVR_VIDEOWALLDISPLAYPOSITION
NET_DVR_WALLOUTPUT_SET	dwCount 个显示输出号(组合)	dwCount 个 NET_DVR_WALLOUTPUTPARAM

[返回目录](#)

6.9.5 电视墙远程控制 **NET_DVR_RemoteControl**

- 函 数: BOOL NET_DVR_RemoteControl(LONG IUserID, DWORD dwCommand, LPVOID lpInBuffer, DWORD dwInBufferSize)
- 参 数:
- [in] IUserID 登录主控板，NET_DVR_Login_V40 返回值
 - [in] dwCommand 控制命令，详见表 6.44
 - [in] lpInBuffer 输入参数，跟控制命令相关，详见列表
 - [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，如表 6.44 所示。

表 6.44 远程控制命令

dwCommand 宏定义	宏定义值	控制功能	lpInBuffer 对应结构体
NET_DVR_CONTROL_SCREEN	1625	屏幕控制	NET_DVR_SCREEN_CONTROL_V41
NET_DVR_DISPLAY_CHANNO_CONTROL	1748	远程控制电视墙上显示编号	NET_DVR_SHOW_CONTROL_INFO
NET_DVR_RESET_VIDEOWALLDISPLAYPOSITION	1750	解除电视墙输出接口绑定	电视墙号(组合): 1 字节电视墙号+3 字节保留

- 设备是否支持以上远程控制能力，可以通过设备能力集进行判断，对应电视墙能力集([WallAbility](#))，相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：WALL_ABILITY，节点分别为：<ScreenControl>、<ShowDispChanNo>、<ResetOutputDisplayPosition>。
- 解除电视墙输出通道绑定，即将所有输出出口的坐标都初始化为 0。

[返回目录](#)

6.9.6 获取 LED 区域信息 [NET_DVR_GetSTDConfig](#)

- 函 数： BOOL NET_DVR_GetSTDConfig(LONG IUserID, DWORD dwCommand, LPNET_DVR_STD_CONFIG lpConfigParam)
- 参 数： [in]IUserID 用户 ID 号，NET_DVR_Login_V40 的返回值
[in]dwCommand 设备配置命令，详见表 6.45
[in&out]lpConfigParam 配置输入输出参数，详见结构体：[NET_DVR_STD_CONFIG](#)，不同的配置功能对应不同的输入输出参数，详见表 6.45
- 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： 获取配置参数时，lpConfigParam 结构体中的 lpInBuffer 无效，设为 NULL。对于不同的配置功能（dwCommand），lpConfigParam 中 lpCondBuffer、lpOutBuffer 分别对应不同的内容，具体如表 6.45 所示。

表 6.45 获取设备参数

dwCommand 宏定义	dwCommand 含义	lpCondBuffer	lpOutBuffer	宏定义值
NET_DVR_GET_LED_AREA_INFO_LIST	获取 LED 区域列表	NET_DVR_LED_AREA_COND	NET_DVR_LED_AREA_INFO_LIST	9295

[返回目录](#)

6.10 电视墙窗口操作

6.10.1 获取窗口参数 [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_V40 的返回值
[in] dwCommand 设备配置命令，详见表 6.46
[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 6.46
[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 6.46 所示。

表 6.46 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_WALLWINPARAM_GET	获取电视墙窗口相关参数	窗口号(组合)	NET_DVR_WALLWINPARAM	9006

[返回目录](#)

6.10.2 设置窗口参数 [NET_DVR_SetDVRConfig](#)

函数: BOOL NET_DVR_SetDVRConfig(LONG lUserID, DWORD dwCommand, LONG IChannel, LPVOID lpInBuffer, DWORD dwInBufferSize)

参数: [in] lUserID 登录主控板，NET_DVR_Login_V40 的返回值
[in] dwCommand 设备配置命令，详见表 6.47
[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 6.47
[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 6.47 所示。

表 6.47 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_WALLWINPARAM_SET	设置电视墙窗口相关参数	窗口号(组合)	NET_DVR_WALLWINPARAM	9005

[返回目录](#)

6.10.3 批量获取窗口信息 [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 登录主控板，NET_DVR_Login_V40 的返回值
[in] dwCommand 设备配置命令，详见表 6.48
[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 6.49
[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 6.49），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

IpOutBuffer 的内容就是无效的
[in] dwOutBufferSize 输出缓冲区大小

表 6.48 批量参数获取命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_GET_VIDEOWALLWINDOWPOSITION	获取电视墙窗口参数	1735

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 IpStatusList[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。IpInBuffer 指定需要获取的 dwCount 个配置信息, IpOutBuffer 保存获取到的 dwCount 个配置信息。
全部获取时 dwCount 置为 0xffffffff, IpInBuffer 设为 1 个电视墙号(组合), dwInBufferSize 置为 4, IpStatusList 置为 NULL, IpOutBuffer 前 4 个字节为返回的个数。如果设置的 IpOutBuffer 缓冲区不足, 仅返回部分信息, 可以根据返回的个数(前 4 字节的值)重新获取。

表 6.49 批量参数获取结构

dwCommand 宏定义	IpInBuffer 对应结构体	IpOutBuffer 对应结构体
NET_DVR_GET_VIDEOWALLWINDOWPOSITION	dwCount 个窗口号(组合)	dwCount 个 NET_DVR_VIDEOWALLWINDOWPOSITION

[返回目录](#)

6.10.4 批量设置窗口 **NET_DVR_SetDeviceConfigEx**

函数: BOOL NET_DVR_SetDeviceConfigEx(LONG iUserID, DWORD dwCommand, DWORD dwCount, NET_DVR_IN_PARAM *IpInParam, NET_DVR_OUT_PARAM *IpOutParam)

参数: [in] iUserID 登录主控板, NET_DVR_Login_V40 的返回值
[in] dwCommand 设备配置命令, 详见表 6.50
[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] IpInParam 输入参数缓冲区, 详见结构体: [NET_DVR_IN_PARAM](#), 其中参数取值根据不同的 dwCommand 而不同, 详见表 6.51
[in] IpOutParam 输出参数缓冲区, 详见结构体: [NET_DVR_OUT_PARAM](#), 其中参数取值根据不同的 dwCommand 而不同, 详见表 6.51

表 6.50 批量配置

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_SET_VIDEOWALLWINDOWPOSITION	设置电视墙窗口参数	1736

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 IpStatusList[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 6.51 批量参数配置结构

dwCommand 宏定义	取值含义
NET_DVR_SET_VIDEOWALLWINDOWPOSITION	设置电视墙窗口参数
输入输出缓冲区参数	参数取值
lpInParam->struCondBuf.pBuf	输入条件缓冲区：新建窗口时为 dwCount 个墙号(组合)，窗口号由设备分配；修改窗口时为 dwCount 个窗口号(组合)。
lpInParam->struCondBuf.nLen	输入条件缓冲区长度
lpInParam->struInParamBuf.pBuf	输入参数缓冲区，dwCount 个 NET_DVR_VIDEOWALLWINDOWPOSITION
lpInParam->struInParamBuf.nLen	输入参数缓冲区长度
lpInParam->dwRecvTimeout	数据接收超时时间，填 0 采用接口默认超时
lpOutParam->struOutBuf.pBuf	输出参数缓冲区，dwCount 个窗口号(组合)
lpOutParam->struOutBuf.nLen	输出参数缓冲区长度
lpOutParam->lpStatusList	状态缓冲区，对应 dwCount 个参数配置的结果，例如 lpStatusList[2]就对应 lpInParam->struInParamBuf.pBuf[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

[返回目录](#)

6.10.5 获取窗口参数配置 **NET_DVR_MatrixGetDecChanCfg**

函 数： BOOL NET_DVR_MatrixGetDecChanCfg(LONG lUserID, DWORD dwDecChan, LPNET_DVR_MATRIX_DECCHAN_CONTROL lpInter)

参 数： [in] lUserID 登录主控板，NET_DVR_Login_V40 返回值
[in] dwDecChan 子窗口号(组合)
[out] lpInter 窗口参数，详见结构体 [NET_DVR_MATRIX_DECCHAN_CONTROL](#)

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](#)

6.10.6 设置窗口参数配置 **NET_DVR_MatrixSetDecChanCfg**

函 数： BOOL NET_DVR_MatrixSetDecChanCfg(LONG lUserID, DWORD dwDecChan, LPNET_DVR_MATRIX_DECCHAN_CONTROL lpInter)

参 数： [in] lUserID 登录主控板，NET_DVR_Login_V40 返回值
[in] dwDecChan 子窗口号(组合)
[in] lpInter 窗口参数，详见结构体 [NET_DVR_MATRIX_DECCHAN_CONTROL](#)

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： 接口里面 dwDecChan 表示子窗口号(组合)：1 字节墙号 + 1 字节子窗口号 + 2 字节窗口号，将子窗口号置为 0xff，窗口号置为 0xffff，表示对所有已上墙的子窗口进行操作。

[返回目录](#)

6.10.7 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 登录主控板, NET_DVR_Login_V40 返回值
[in] dwCommand 控制命令, 详见表 6.52
[in] lpInBuffer 输入参数, 跟控制命令相关, 详见列表
[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, 如表 6.52 所示。

表 6.52 远程控制命令

dwCommand 宏定义	宏定义值	控制功能	lpInBuffer 对应结构体
NET_DVR_VIDEOWALLWINDOW_CLOSEALL	1737	电视墙关闭所有窗口	4 字节, 窗口号(组合)
NET_DVR_SWITCH_WIN_TOP	9017	窗口置顶(即窗口图层号设为最大)	4 字节, 窗口号(组合)
NET_DVR_SWITCH_WIN_BOTTOM	9018	窗口置底(即窗口图层号设为最小)	4 字节, 窗口号(组合)

[返回目录](#)

6.11 窗口解码操作

主动解码

6.11.1 启动动态解码 **NET_DVR_MatrixStartDynamic_V41**

函 数: BOOL NET_DVR_MatrixStartDynamic_V41(LONG IUserID, DWORD dwDecChanNum, LPNET_DVR_PU_STREAM_CFG_V41 lpDynamicInfo)

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V40 返回值
[in] dwDecChanNum 子窗口号(组合)
[in] lpDynamicInfo 动态解码参数, 详见结构体 [NET_DVR_PU_STREAM_CFG_V41](#)

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](#)

6.11.2 停止动态解码 **NET_DVR_MatrixStopDynamic**

函 数: BOOL NET_DVR_MatrixStopDynamic(LONG IUserID, DWORD dwDecChanNum)

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V40 返回值
[in] dwDecChanNum 子窗口号(组合)

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](#)

6.11.3 批量设置动态解码 **NET_DVR_SetDeviceConfig**

- 函 数: BOOL NET_DVR_SetDeviceConfig(LONG IUserID,DWORD dwCommand,DWORD dwCount,LPVOID lpInBuffer,DWORD dwInBufferSize,LPVOID lpStatusList, LPVOID lpInParamBuffer,DWORD dwInParamBufferSize)
- 参 数:
- [in] IUserID 登录主控板, NET_DVR_Login_V40 的返回值
 - [in] dwCommand 设备配置命令, 详见表 6.53
 - [in] dwCount 要设置的子设备个数
 - [in] lpInBuffer 配置条件缓冲区, 详见表 6.54
 - [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
 - [in] lpInParamBuffer 需要设置给设备的参数内容 (详见表 6.54), 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, 表示对应的 lpInBuffer 设置失败, 为 0 则设置成功
 - [in] dwInParamBufferSize 设置内容缓冲区大小

表 6.53 批量参数配置命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_DYNAMIC_DECODE_BATCH	批量动态解码	1769

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。lpInBuffer 指定需要设置的 dwCount 个监控点信息, lpOutBuffer 保存将要设置的 dwCount 个监控点的配置信息。批量配置动态解码, 当设备判断某些配置错误时, 返回相应错误状态值, 子窗口动态解码情况不变, 而对成功配置的子窗口启动动态解码。

表 6.54 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_DYNAMIC_DECODE_BATCH	dwCount 个 4 字节子窗口号(组合)	dwCount 个 NET_DVR_PU_STREAM_CFG_V41

[返回目录](#)

6.11.4 获取轮巡解码通道 **NET_DVR_MatrixGetLoopDecChanInfo_V41**

- 函 数: BOOL NET_DVR_MatrixGetLoopDecChanInfo_V41(LONG IUserID, DWORD dwDecChanNum, LPNET_DVR_MATRIX_LOOP_DECINFO_V41 lpOuter)
- 参 数:
- [in] IUserID 登录主控板, NET_DVR_Login_V40 返回值
 - [in] dwDecChanNum 子窗口号(组合)
 - [out] lpOuter 轮巡的解码通道参数, 详见结构体 [NET_DVR_MATRIX_LOOP_DECINFO_V41](#)
- 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](#)

6.11.5 设置轮巡解码通道 **NET_DVR_MatrixSetLoopDecChanInfo_V41**

函 数: BOOL NET_DVR_MatrixSetLoopDecChanInfo_V41(LONG IUserID, DWORD dwDecChanNum, LPNET_DVR_MATRIX_LOOP_DECINFO_V41 lpInter)

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V40 返回值
[in] dwDecChanNum 子窗口号(组合)
[in] lpInter 轮巡的解码通道参数, 详见结构体

[NET_DVR_MATRIX_LOOP_DECINFO_V41](#)

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](#)

6.11.6 远程文件回放解码 **NET_DVR_RemoteControl**

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

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V40 返回值
[in] dwCommand 控制命令, 详见表 6.55
[in] lpInBuffer 输入参数, 跟控制命令相关, 详见表 6.55
[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, 如表 6.55 所示。

表 6.55 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_DEC_PLAY_REMOTE_FILE	9032	解码播放远程文件	NET_DVR_MATRIX_DEC_REMOTE_PLAY_EX

[返回目录](#)

6.11.7 远程回放文件控制 **NET_DVR_MatrixSetRemotePlayControl**

函 数: BOOL NET_DVR_MatrixSetRemotePlayControl(LONG IUserID, DWORD dwDecChanNum, DWORD dwControlCode, DWORD dwInValue, DWORD *lpOutValue)

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V40 返回值
[in] dwDecChanNum 子窗口号(组合)
[in] dwControlCode 控制命令, 具体定义见表 6.56
[in] dwInValue 设置参数
[out] lpOutValue 获取到的参数指针

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: 接口中的 dwInValue 和 lpOutValue 参数根据不同的命令号决定是否输入和输出, 如当进行 NET_DVR_PLAYSETPOS 命令操作时, 需要对 dwInValue 参数进行赋值。

表 6.56 回放控制命令

宏定义	宏定义值	含义
NET_DVR_PLAYSTART	1	开始播放
NET_DVR_PLAYSTOP	2	停止播放
NET_DVR_PLAYPAUSE	3	暂停播放
NET_DVR_PLAYRESTART	4	恢复播放
NET_DVR_PLAYFAST	5	快放
NET_DVR_PLAYSLOW	6	慢放
NET_DVR_PLAYNORMAL	7	正常速度播放
NET_DVR_PLAYSTARTAUDIO	9	打开声音
NET_DVR_PLAYSTOPAUDIO	10	关闭声音
NET_DVR_PLAYSETPOS	12	改变文件回放的进度

[返回目录](#)

6.11.8 获取回放状态 **NET_DVR_MatrixGetRemotePlayStatus**

函 数: BOOL NET_DVR_MatrixGetRemotePlayStatus(LONG IUserID, DWORD dwDecChanNum, LPNET_DVR_MATRIX_DEC_REMOTE_PLAY_STATUS lpOuter)

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V40 返回值
[in] dwDecChanNum 子窗口号(组合)
[out] lpOuter 回放状态, 详见结构体

[NET_DVR_MATRIX_DEC_REMOTE_PLAY_STATUS](#)

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: 按时间回放不支持进度控制; 由于回放控制命令是转发实现, 存在一定的延迟, 因此, 回放控制命令不宜过于频繁调用, 具体视网络状况而定, 当把获取的状态作为客户端处理依据时应考虑网络转发的延迟因素; 按时间回放时, 获取回放状态所得到的文件信息是当前播放的单个片段的信息, 并非整个时间范围内全部片段的信息, 判断播放是否结束使用 NET_DVR_MATRIX_DEC_REMOTE_PLAY_STATUS 结构中的 dwCurDataType 成员。

[返回目录](#)

解码开关控制

6.11.9 获取窗口解码开关 **NET_DVR_MatrixGetDecChanEnable**

函 数: BOOL NET_DVR_MatrixGetDecChanEnable(LONG IUserID, DWORD dwDecChanNum, LPDWORD lpdwEnable)

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V40 返回值
[in] dwDecChanNum 子窗口号(组合)
[out] lpdwEnable 0 表示关闭, 1 表示打开

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：窗口的解码开关，用于控制窗口的解码过程，设置此开关为关时，无论当前窗口是处在动态解码还是轮巡过程，都将停止解码，生效后该通道处于黑屏状态或者最后一帧画面；设置开关为开时，将恢复先前过程。此功能可与轮巡，轮巡开关等配合使用。

[返回目录](#)

6.11.10 设置窗口解码开关 **NET_DVR_MatrixSetDecChanEnable**

函数：BOOL NET_DVR_MatrixSetDecChanEnable (LONG IUserID, DWORD dwDecChanNum, DWORD dwEnable)

参数：
 [in] IUserID 登录主控板，NET_DVR_Login_V40 返回值
 [in] dwDecChanNum 子窗口号(组合)
 [in] dwEnable 0 表示关闭，1 表示打开

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：窗口的解码开关，用于控制窗口的解码过程，设置此开关为关时，无论当前窗口是处在动态解码还是轮巡过程，都将停止解码，生效后该通道处于黑屏状态或者最后一帧画面（详见：[NET_DVR_WALL_CFG](#)）；设置开关为开时，将恢复先前过程。此功能可与轮巡，轮巡开关等配合使用。将子窗口号(组合)中的子窗口号填为 0xff，窗口号填为 0xffff，表示对某个墙的所有窗口进行解码开关控制。

[返回目录](#)

6.11.11 批量获取解码通道开关 **NET_DVR_GetDeviceConfig**

函数：BOOL NET_DVR_GetDeviceConfig(LONG IUserID, DWORD dwCommand, DWORD dwCount, LPVOID lpInBuffer, DWORD dwInBufferSize, LPVOID lpStatusList, LPVOID lpOutBuffer, DWORD dwOutBufferSize)

参数：
 [in] IUserID 登录主控板，NET_DVR_Login_V40 的返回值
 [in] dwCommand 设备配置命令，详见表 6.57
 [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 6.58
 [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 6.58），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 6.57 批量参数获取命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_DEC SWITCH_GET_BATCH	批量获取解码通道开关	1771

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。

因。

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

表 6.58 批量参数获取结构

dwCommand 宏定义	IpInBuffer 对应结构体	IpOutBuffer 对应结构体
NET_DVR_DECSWITCH_GET_BATCH	dwCount 个 4 字节子窗口号(组合)	dwCount 个 4 字节操作命令(0-关, 1-开)

- 对不存在的或没有动态解码的子窗口，调用批量获取或设置，对应状态值返回对应的错误码。

[返回目录](#)

6.11.12 批量设置解码通道开关 **NET_DVR_SetDeviceConfig**

函数： BOOL NET_DVR_SetDeviceConfig(LONG IUserID,DWORD dwCommand,DWORD dwCount,LPVOID IpInBuffer,DWORD dwInBufferSize,LPVOID IpStatusList, LPVOID IpInParamBuffer,DWORD dwInParamBufferSize)

参数：

[in] IUserID	登录主控板，NET_DVR_Login_V40 的返回值
[in] dwCommand	设备配置命令，详见表 6.59
[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] IpInBuffer	配置条件缓冲区，详见表 6.60
[in] dwInBufferSize	缓冲区长度
[out] IpStatusList	错误信息列表，和要查询的监控点一一对应，例如 IpStatusList[2] 就对应 IpInBuffer[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
[in] IpInParamBuffer	需要设置给设备的参数内容（详见表 6.60），和要查询的监控点一一对应。如果某个监控点对应的 IpStatusList 信息为大于 0 值，表示对应的 IpInBuffer 设置失败，为 0 则设置成功
[in] dwInParamBufferSize	设置内容缓冲区大小

表 6.59 批量参数配置命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_DECSWITCH_SET_BATCH	批量设置解码通道开关	1770

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 IpStatusList[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。IpInBuffer 指定需要设置的 dwCount 个监控点信息，IpOutBuffer 保存将要设置的 dwCount 个监控点的配置信息。

表 6.60 批量参数配置结构

dwCommand 宏定义	IpInBuffer 对应结构体	IpInParamBuffer 对应结构体
NET_DVR_DECSWITCH_SET_BATCH	dwCount 个 4 字节子窗口号(组合)	dwCount 个 4 字节操作命令(0-关, 1-开)

- 对不存在的或没有动态解码的子窗口，调用批量获取或设置，对应状态值返回对应的错误码。

[返回目录](#)

6.11.13 获取窗口轮巡解码开关 **NET_DVR_MatrixGetLoopDecChanEnable**

函 数： BOOL NET_DVR_MatrixGetLoopDecChanEnable(LONG IUserID, DWORD dwDecChanNum, LPDWORD lpdwEnable)

参 数： [in] IUserID 登录主控板，NET_DVR_Login_V40 返回值
[in] dwDecChanNum 子窗口号(组合)
[out] lpdwEnable 0 表示关闭，1 表示打开

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](#)

6.11.14 设置窗口轮巡解码开关 **NET_DVR_MatrixSetLoopDecChanEnable**

函 数： BOOL NET_DVR_MatrixSetLoopDecChanEnable(LONG IUserID, DWORD dwDecChanNum, DWORD dwEnable)

参 数： [in] IUserID 登录主控板，NET_DVR_Login_V40 返回值
[in] dwDecChanNum 子窗口号(组合)
[in] dwEnable 0 表示关闭，1 表示打开

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： 轮巡开关用于控制轮巡过程的启停，而不是控制解码的启停，当设置当前正在轮巡解码的窗口的轮巡开关为关时，该窗口停止循环，停留在当前所连接的输入源继续解码，退为动态解码，当设置轮巡开关为开时，设备恢复该窗口的循环。

[返回目录](#)

被动解码

6.11.15 启动被动解码 **NET_DVR_MatrixStartPassiveDecode**

函 数： LONG NET_DVR_MatrixStartPassiveDecode(LONG IUserID, DWORD dwDecChanNum, LPNET_DVR_MATRIX_PASSIVEMODE lpPassiveMode)

参 数： [in] IUserID 登录主控板，NET_DVR_Login_V40 返回值
[in] dwDecChanNum 子窗口号(组合)
[in] lpPassiveMode 被动解码参数，详见结构体 [NET_DVR_MATRIX_PASSIVEMODE](#)

返回值： -1 表示失败，其他值作为 NET_DVR_MatrixSendData 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](#)

6.11.16 向被动解码窗口发送数据 **NET_DVR_MatrixSendData**

函 数： BOOL NET_DVR_MatrixSendData(LONG IPassiveHandle, char *pSendBuf, DWORD dwBufSize)

参 数： [in] IPassiveHandle NET_DVR_MatrixStartPassiveDecode 的返回值

[in] pSendBuf 发送数据缓冲区

[in] dwBufSize 发送缓冲区大小，需小于 256K 字节，推荐 30K 字节

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](#)

6.11.17 停止被动解码 **NET_DVR_MatrixStopPassiveDecode**

函 数: BOOL NET_DVR_MatrixStopPassiveDecode(LONG lPassiveHandle)

参 数: [in] lPassiveHandle NET_DVR_MatrixStartPassiveDecode 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](#)

6.11.18 获取被动解码状态 **NET_DVR_MatrixGetPassiveDecodeStatus**

函 数: LONG NET_DVR_MatrixGetPassiveDecodeStatus(LONG lPassiveHandle)

参 数: [in] lPassiveHandle NET_DVR_MatrixStartPassiveDecode 的返回值

返回值: -1 表示失败，1- 发送成功，2- 暂停发送，3- 恢复发送，4- 错误，5- 心跳信息。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码可以判断出错原因。

说 明:

[返回目录](#)

6.11.19 被动解码播放控制 **NET_DVR_MatrixPassiveDecodeControl**

函 数: BOOL NET_DVR_MatrixPassiveDecodeControl(LONG lUserID, DWORD dwDecChanNum, LPNET_DVR_PASSIVEDECODE_CONTROL lpInter)

参 数: [in] lUserID 登录主控板，NET_DVR_Login_V40 返回值

[in] dwDecChanNum 子窗口号(组合)

[in] lpInter 解码控制参数，详见结构体 [NET_DVR_PASSIVEDECODE_CONTROL](#)

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](#)

窗口解码信息

6.11.20 获取当前窗口解码信息 **NET_DVR_MatrixGetDecChanInfo_V41**

函 数: BOOL NET_DVR_MatrixGetDecChanInfo_V41(LONG lUserID, DWORD dwDecChanNum, LPNET_DVR_MATRIX_DEC_CHAN_INFO_V41 lpOuter)

参 数: [in] lUserID 登录主控板，NET_DVR_Login_V40 的返回值

[in] dwDecChanNum 子窗口号(组合)
[out] lpOuter 窗口的解码信息, 详见结构体

[NET_DVR_MATRIX_DEC_CHAN_INFO_V41](#)

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: 获取当前窗口的解码信息, 包括前端编码设备通道信息、取流模式、解码状态等。

[返回目录](#)

6.11.21 批量获取窗口解码信息 **NET_DVR_GetDeviceConfig**

函数: BOOL NET_DVR_GetDeviceConfig(LONG IUserID, DWORD dwCommand, DWORD dwCount, LPVOID lpInBuffer, DWORD dwInBufferSize, LPVOID lpStatusList, LPVOID lpOutBuffer, DWORD dwOutBufferSize)

参数: [in] IUserID 登录主控板, NET_DVR_Login_V40 的返回值
[in] dwCommand 设备配置命令, 详见表 6.61
[in] dwCount 要设置的监控点个数, 0 和 1 都表示 1 个监控点信息
[in] lpInBuffer 配置条件缓冲区, 详见表 6.62
[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 6.62), 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 6.61 批量参数获取命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_GET_WIN_DEC_INFO	获取窗口解码信息	1749
NET_DVR_GET_WINASSOCIATEDDEVINFO	获取电视墙窗口对应资源信息	9010

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监控点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。

全部获取时 dwCount 置为 0xffffffff, lpInBuffer 置为 NULL, dwInBufferSize 置为 0, lpStatusList 置为 NULL, lpOutBuffer 前 4 个字节为返回的个数。如果设置的 lpOutBuffer 缓冲区不足, 仅返回部分信息, 可以根据返回的个数 (前 4 字节的值) 重新获取。其中, **NET_DVR_GET_WIN_DEC_INFO** 不支持全部获取。

表 6.62 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_WIN_DEC_INFO	dwCount 个子窗口号(组合)	dwCount 个 NET_DVR_MATRIX_DEC_CHAN_INFO_V41
NET_DVR_GET_WINASSOCIATEDDEVINFO	dwCount 个 NET_DVR_WININDEX_INFO	dwCount 个 NET_DVR_ASSOCIATEDDEV_CHAN_INFO

[返回目录](#)

6.11.22 获取窗口解码状态信息 **NET_DVR_GetDeviceStatus**

- 函 数：** BOOL NET_DVR_GetDeviceStatus(LONG lUserID,DWORD dwCommand,DWORD dwCount, LPVOID lpInBuffer, DWORD dwInBufferSize,LPVOID lpStatusList, LPVOID lpOutBuffer, DWORD dwOutBufferSize)
- 参 数：**
- [in] lUserID 登录主控板，NET_DVR_Login_V40 返回值
 - [in] dwCommand 设备配置命令，详见表 6.63
 - [in] dwCount 批量获取个数，0 和 1 都表示获取 1 个设备信息
 - [in] lpInBuffer 配置条件缓冲区，详见表 6.63。只获取一个设备的状态时，该参数可设为 NULL
 - [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 (1 个 32 位无符号整数值)，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 6.63），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。如果只获取一个设备的状态且 lpInBuffer 设为 NULL 时，该参数可设为 NULL
 - [in] dwOutBufferSize 输出缓冲区大小
- 返回值：** TRUE 表示成功，但不代表每一个状态获取都成功，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设备状态信息的通用接口。不同的获取功能对应不同的结构体和命令号，如表 6.63 所示。

表 6.63 解码状态信息

dwCommand 宏定义	dwCommand 含义	lpInBuffer 对应结构体	lpOutBuffer 对应结构体	宏定义值
NET_DVR_MATRIX_GETWINSTATUS	电视墙中获取窗口解码状态	dwCount 个 NET_DVR_WALLWIN_INFO	dwCount 个 NET_DVR_WALL_WIN_STATUS	9009

[返回目录](#)

电子放大

6.11.23 大屏电子放大 **NET_DVR_ScreenZoomIn**

- 函 数：** BOOL NET_DVR_ScreenZoomIn(LONG lUserID, LPNET_DVR_SCREENZOOM pStruScreen)
- 参 数：**
- [in] lUserID 登录主控板，NET_DVR_Login_V40 的返回值
 - [in] pStruScreen 大屏放大信息，详见结构体 [NET_DVR_SCREENZOOM](#)
- 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：** 用于设置大屏区域放大或缩小。

[返回目录](#)

JPEG 抓图

6.11.24 单帧数据捕获并保存成 JPEG 图片 **NET_DVR_CaptureJPEGPicture**

函 数： BOOL NET_DVR_CaptureJPEGPicture(LONG IUserID, LONG IChannel, LPNET_DVR_JPEGPARA lpJpegPara, char *sPicFileName)

参 数： [in]IUserID 登录主控板，NET_DVR_Login_V40 的返回值
[in]IChannel 子窗口号(组合)
[in]lpJpegPara JPEG 图像参数，详见结构体 [NET_DVR_JPEGPARA](#)
[in]sPicFileName 保存 JPEG 图的文件路径

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： 该接口用于设备的单帧数据捕获，并保存成 JPEG 图片。IPC 设备支持当前视频分辨率的抓取。

[返回目录](#)

6.11.25 单帧数据捕获并保存成 JPEG 存放在指定的内存空间中

NET_DVR_CaptureJPEGPicture_NEW

函 数： BOOL NET_DVR_CaptureJPEGPicture_NEW(LONG IUserID, LONG IChannel, LPNET_DVR_JPEGPARA lpJpegPara, char *sJpegPicBuffer, DWORD dwPicSize, LPDWORD lpSizeReturned)

参 数： [in]IUserID 登录主控板，NET_DVR_Login_V40 的返回值
[in]IChannel 子窗口号(组合)
[in]lpJpegPara JPEG 图像参数，详见结构体 [NET_DVR_JPEGPARA](#)
[in]sJpegPicBuffer 保存 JPEG 数据的缓冲区
[in]dwPicSize 输入缓冲区大小，不能小于图片数据的大小
[out]lpSizeReturned 返回图片数据的大小

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： 该接口用于设备的单帧数据捕获，并保存成 JPEG 图片。IPC 设备支持当前视频分辨率的抓取。

[返回目录](#)

6.12 音频切换

6.12.1 批量获取音频切换参数 **NET_DVR_GetDeviceConfig**

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

参 数： [in] IUserID 登录主控板，NET_DVR_Login_V40 的返回值
[in] dwCommand 设备配置命令，详见表 6.64
[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 6.65

[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 6.65），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 6.64 批量参数获取命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_GET_VW_AUDIO_CFG	获取音频切换参数	1753

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监控点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 6.65 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_VW_AUDIO_CFG	dwCount 个 NET_DVR_AUDIO_CHAN_INFO	dwCount 个 NET_DVR_AUDIO_CHAN_CFG

[返回目录](#)

6.12.2 批量设置音频切换参数 **NET_DVR_SetDeviceConfig**

函数：BOOL NET_DVR_SetDeviceConfig(LONG lUserID,DWORD dwCommand,DWORD dwCount,LPVOID lpInBuffer,DWORD dwInBufferSize,LPVOID lpStatusList, LPVOID lpInParamBuffer,DWORD dwInParamBufferSize)

参数：

[in] lUserID	登录主控板，NET_DVR_Login_V40 的返回值
[in] dwCommand	设备配置命令，详见表 6.66
[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] lpInBuffer	配置条件缓冲区，详见表 6.67
[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
[in] lpInParamBuffer	需要设置给设备的参数内容（详见表 6.67），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，表示对应的 lpInBuffer 设置失败，为 0 则设置成功
[in] dwInParamBufferSize	设置内容缓冲区大小

表 6.66 批量参数配置命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_SET_VW_AUDIO_CFG	设置音频切换参数	1752

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。IpInBuffer 指定需要设置的 dwCount 个监控点信息，IpOutBuffer 保存将要设置的 dwCount 个监控点的配置信息。

表 6.67 批量参数配置结构

dwCommand 宏定义	IpInBuffer 对应结构体	IpInParamBuffer 对应结构体
NET_DVR_SET_VW_AUDIO_CFG	dwCount 个 NET_DVR_AUDIO_CHAN_INFO	dwCount 个 NET_DVR_AUDIO_CHAN_CFG

[返回目录](#)

6.13 场景功能

配置和控制

6.13.1 批量获取场景参数 **NET_DVR_GetDeviceConfig**

函数： BOOL NET_DVR_GetDeviceConfig(LONG lUserID, DWORD dwCommand, DWORD dwCount, LPVOID IpInBuffer, DWORD dwInBufferSize, LPVOID IpStatusList, LPVOID IpOutBuffer, DWORD dwOutBufferSize)

参数：

- [in] lUserID 登录主控板，NET_DVR_Login_V40 的返回值
- [in] dwCommand 设备配置命令，详见表 6.68
- [in] dwCount 要设置的监控点个数，0 和 1 都表示 1 个监控点信息
- [in] IpInBuffer 配置条件缓冲区，详见表 6.69
- [in] dwInBufferSize 缓冲区长度
- [out] IpStatusList 错误信息列表，和要查询的监控点一一对应，例如 IpStatusList[2] 就对应 IpInBuffer[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] IpOutBuffer 设备返回的参数内容（详见表 6.69），和要查询的监控点一一对应。如果某个监控点对应的 IpStatusList 信息为大于 0 值，对应 IpOutBuffer 的内容就是无效的
- [in] dwOutBufferSize 输出缓冲区大小

表 6.68 批量参数获取命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_GET_CURRENT_SCENE	获取当前场景号	1745
NET_DVR_GET_VW_SCENE_PARAM	获取电视墙场景模式参数	1746

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 IpStatusList[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： 该该接口是带有发送数据的批量获取监控点配置信息的通用接口。IpInBuffer 指定需要获取的 dwCount 个配置信息，IpOutBuffer 保存获取得到的 dwCount 个配置信息。
全部获取时 dwCount 置为 0xffffffff，IpInBuffer 置为 NULL，dwInBufferSize 置为 0，IpStatusList 置为 NULL，IpOutBuffer 前 4 个字节为返回的个数。如果设置的 IpOutBuffer 缓冲区不足，仅返回部

分信息，可以根据返回的个数（前 4 字节的值）重新获取。

表 6.69 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_CURRENT_SCENE	dwCount 个 NET_DVR_VIDEO_WALL_INFO	dwCount 个 4 字节场景号
NET_DVR_GET_VW_SCENE_PARAM	dwCount 个 NET_DVR_VIDEO_WALL_INFO	dwCount 个 NET_DVR_WALLSCENECFG

[返回目录](#)

6.13.2 批量设置场景参数 **NET_DVR_SetDeviceConfig**

函 数： BOOL NET_DVR_SetDeviceConfig(LONG IUserID,DWORD dwCommand,DWORD dwCount,LPVOID lpInBuffer,DWORD dwInBufferSize,LPVOID lpStatusList, LPVOID lpInParamBuffer,DWORD dwInParamBufferSize)

参 数：

- [in] IUserID 登录主控板，NET_DVR_Login_V40 的返回值
- [in] dwCommand 设备配置命令，详见表 6.70
- [in] dwCount 要设置的子设备个数
- [in] lpInBuffer 配置条件缓冲区，详见表 6.71
- [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
- [in] lpInParamBuffer 需要设置给设备的参数内容（详见表 6.71），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，表示对应的 lpInBuffer 设置失败，为 0 则设置成功
- [in] dwInParamBufferSize 设置内容缓冲区大小

表 6.70 批量参数配置命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_SET_VW_SCENE_PARAM	设置电视墙场景模式参数	1747

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。lpInBuffer 指定需要设置的 dwCount 个监控点信息，lpOutBuffer 保存将要设置的 dwCount 个监控点的配置信息。

表 6.71 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_VW_SCENE_PARAM	dwCount 个 NET_DVR_VIDEO_WALL_INFO	dwCount 个 NET_DVR_WALLSCENECFG

[返回目录](#)

6.13.3 场景切换控制 **NET_DVR_MatrixSceneControl**

函 数： BOOL NET_DVR_MatrixSceneControl(LONG IUserID, DWORD dwSceneNum, DWORD dwCmd, DWORD dwCmdParam)

参 数： [in] IUserID 登录主控板，NET_DVR_Login_V40 的返回值

[in] dwSceneNum 场景号，能力集中获取

[in] dwCmd 命令类型：

- 1- 场景模式切换（如果要切换的是当前场景，则不进行切换）；
- 2- 初始化场景（将此场景的配置清空，如果是当前场景，则同时对当前场景进行清屏操作）；3- 强制切换（无论是否是当前场景，强制切换）；4- 保存当前模式到某场景

[in] dwCmdParam 命令参数，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：

[返回目录](#)

6.13.4 远程控制场景 **NET_DVR_RemoteControl**

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

参 数： [in] IUserID 登录主控板，NET_DVR_Login_V40 返回值

[in] dwCommand 控制命令，详见表 6.72

[in] lpInBuffer 输入参数，跟控制命令相关，详见列表

[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，如表 6.72 所示。

表 6.72 远程控制命令

dwCommand 宏定义	宏定义值	控制功能	lpInBuffer 对应结构体
NET_DVR_SCENE_CONTROL	1744	场景控制	NET_DVR_SCENE_CONTROL_INFO

[返回目录](#)

6.13.5 获取当前正在使用的场景模式 **NET_DVR_MatrixGetCurrentSceneMode**

函 数： BOOL NET_DVR_MatrixGetCurrentSceneMode(LONG IUserID, DWORD *dwSceneNum)

参 数： [in] IUserID 登录主控板，NET_DVR_Login_V40 的返回值

[out] dwSceneNum 场景号

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：

[返回目录](#)

场景配置文件导入导出

6.13.6 导入场景配置文件 **NET_DVR_UploadFile**

函 数： LONG NET_DVR_UploadFile(LONG IUserID, DWORD dwUploadType, LPVOID lpInBuffer, DWORD

dwInBufferSize, char *sFileName)

- 参 数:
- [in] IUserID 登录主控板, NET_DVR_Login_V40 的返回值
 - [in] dwUploadType 上传文件类型, 详见表 6.73
 - [in] lpInBuffer 不同的 dwUploadType, 输入参数不同, 详见表 6.73
 - [in] dwInBufferSize 输入缓冲区大小
 - [in] *sFileName 上传文件的绝对路径(包括文件名), 最多 128 字节, 文件名最多 32 字节(含最后的'\0')

表 6.73 上传文件类型

dwUploadType 宏定义	宏定义值	dwUploadType 含义	lpInBuffer 对应结构体
SCENE_CONFIGURATION_FILE	5	上传场景配置文件	NET_DVR_VIDEO_WALL_INFO

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](#)

6.13.7 获取文件导入的进度和状态 **NET_DVR_GetUploadState**

函 数: LONG NET_DVR_GetUploadState(LONG IUploadHandle, LPDWORD pProgress)

- 参 数:
- [in] IUploadHandle 文件上传的句柄, NET_DVR_UploadFile 的返回值
 - [out] pProgress 返回的进度值, 取值范围: 0~100

返回值: -1 表示函数调用失败, 其他为上传的状态值: 1- 上传成功; 2- 正在上传; 3- 上传失败; 4- 网络断开, 状态未知。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](#)

6.13.8 停止导入配置文件 **NET_DVR_UploadClose**

函 数: BOOL NET_DVR_UploadClose(LONG IUploadHandle)

- 参 数:
- [in] IUploadHandle 文件上传的句柄, NET_DVR_UploadFile 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](#)

6.13.9 导出场景配置文件 **NET_DVR_StartDownload**

函 数: LONG NET_DVR_StartDownload(LONG IUserID, DWORD dwDownloadType, LPVOID lpInBuffer, DWORD dwInBufferSize, char const *sFileName)

- 参 数:
- [in] IUserID 登录主控板, NET_DVR_Login_V40 的返回值
 - [in] dwDownloadType 下载文件类型, 详见表 6.74
 - [in] lpInBuffer 输入参数。不同的 dwUploadType, 输入参数不同, 详见表 6.74
 - [in] dwInBufferSize 输入缓冲区大小
 - [in] sFileName 下载文件的保存路径(绝对路径, 包括文件名)

返回值: -1 表示失败,其他值作为 NET_DVR_StopDownload 和 NET_DVR_GetDownloadState 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:

表 6.74 文件下载类型

dwDownloadType 宏定义	宏定义值	dwDownloadType 含义	lpInBuffer 对应结构体
NET_SDK_DOWNLOAD_SCENE_CONFIGURATION_FILE	5	下载场景配置文件	NET_DVR_VIDEO_WALL_INFO

[返回目录](#)

6.13.10 获取文件导出的进度和状态 **NET_DVR_GetDownloadState**

函 数: LONG NET_DVR_GetDownloadState(LONG IDownloadHandle, LPDWORD pProgress)

参 数: [in] IDownloadHandle 文件下载的句柄, NET_DVR_StartDownload 的返回值
[out] pProgress 返回的进度值,取值范围: 0~100

返回值: -1 表示函数调用失败,其他为下载的状态值: 1- 下载成功; 2- 正在下载; 3- 下载失败; 4- 网络断开, 状态未知。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](#)

6.13.11 停止导出配置文件 **NET_DVR_StopDownload**

函 数: BOOL NET_DVR_StopDownload(LONG IHandle);

参 数: [in] IHandle 文件下载的句柄, NET_DVR_StartDownload 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](#)

6.14 预案功能

6.14.1 获取预案列表 **NET_DVR_GetPlanList**

函 数: BOOL NET_DVR_GetPlanList(LONG IUserID, DWORD dwDevNum, LPNET_DVR_PLAN_LIST lpPlanList)

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V40 返回值

[in] dwDevNum 保留, 设为 0

[out] lpPlanList 预案列表, 详见结构体 [NET_DVR_PLAN_LIST](#)

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:

- 1.调用该接口, 将 [NET_DVR_PLAN_LIST](#) 中的 pBuffer 置为空且 dwBufLen 为 0, 此时获取预案个数 (dwPlanNums);
- 2.根据第一次调用接口返回的数量分配 pBuffer 的大小, 然后再次调用该接口, 此时获取预案列表 (pBuffer)。

[返回目录](#)

6.14.2 远程控制预案 **NET_DVR_RemoteControl**

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

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V40 返回值
[in] dwCommand 控制命令, 详见表 6.75
[in] lpInBuffer 输入参数, 跟控制命令相关, 详见列表
[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, 如表 6.75 所示。

表 6.75 远程控制命令

dwCommand 宏定义	宏定义值	控制功能	lpInBuffer 对应结构体
NET_DVR_CTRL_PLAN	1617	控制预案 (开始或停止)	NET_DVR_CONTROL_PARAM

[返回目录](#)

6.14.3 获取预案信息 **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_V40 的返回值
[in] dwCommand 设备配置命令, 详见表 6.76
[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 6.76
[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 6.76 所示。

如果在获取正在执行预案号时没有执行预案, 返回预案号为 0。设备是否支持获取正在执行的预案信息能力, 可以通过设备能力集进行判断, 对应电视墙能力集 ([WallAbility](#)), 相关接口:

[NET_DVR_GetDeviceAbility](#), 能力集类型: WALL_ABILITY, 节点: <SupportGetPlayingPlan>。

表 6.76 获取预案信息

dwCommand 宏定义	dwCommand 含义	IChannel	lpOutBuffer 对应结构体	宏定义值
NET_DVR_GET_PLAYING_PLAN	获取正在执行的预案状态	电视墙号(组合)	一个 4 字节正在执行的预案号	1773

[返回目录](#)

6.14.4 设置预案 **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_V40 的返回值
[in] dwCommand 设备配置命令, 详见表 6.77

[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 6.77

[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 6.77 所示。

表 6.77 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_PLAN	设置预案	预案号，从 1 开始	NET_DVR_PLAN_CFG	1616

[返回目录](#)

6.15 底图配置、上传和控制

6.15.1 获取底图参数 [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_V40 的返回值

[in] dwCommand 设备配置命令，详见表 6.78

[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 6.78

[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 6.78 所示。

表 6.78 底图参数获取

dwCommand 宏定义	dwCommand 含义	IChannel	lpOutBuffer 对应结构体	宏定义值
NET_DVR_GET_BASEMAP_CFG	获取底图参数	底图窗口号(组合)	NET_DVR_BASEMAP_CONTROL_CFG	9121
NET_DVR_GET_BASEMAP_WIN_CFG	获取底图窗口参数	底图窗口号(组合)	NET_DVR_BASEMAP_WINCFG	9124
NET_DVR_GET_BASEMAP_PIC_INFO	获取底图图片信息	底图序号	NET_DVR_BASEMAP_PIC_INFO	9127

[返回目录](#)

6.15.2 设置底图参数 [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_V40 的返回值

[in] dwCommand 设备配置命令，详见表 6.79

[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 6.79

[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 6.79 所示。

表 6.79 底图参数设置

dwCommand 宏定义	dwCommand 含义	IChannel	lpInBuffer 对应结构体	宏定义值
NET_DVR_SET_BASEMAP_CFG	设置底图参数	底图窗口号(组合)	NET_DVR_BASEMAP_CONTROL_CFG	9120
NET_DVR_SET_BASEMAP_WIN_CFG	设置底图窗口参数	底图窗口号(组合)	NET_DVR_BASEMAP_WINCFG	9125

[返回目录](#)

6.15.3 图片上传 **NET_DVR_PicUpload**

函数: LONG NET_DVR_PicUpload(LONG IUserID, char const *sFileName, LPNET_DVR_PICTURECFG lpPictureCfg)

参数: [in] IUserID 登录主控板, NET_DVR_Login_V40 的返回值

[in] sFileName 上传图片的图片路径

[in] lpPictureCfg 图片参数, 详见结构体 [NET_DVR_PICTURECFG](#)

返回值: -1 表示失败, 其他值作为 NET_DVR_GetPicUploadState 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: B20 上传底图不做切割, 直接一次发送。底图要求 jpg 格式, 宽高必须 16 对齐, 支持分辨率: 1280*720 ~ 7680 * 4320。

[返回目录](#)

6.15.4 获取图片上传的进度 **NET_DVR_GetPicUploadProgress**

函数: LONG NET_DVR_GetPicUploadProgress(LONG IUploadHandle)

参数: [in] IUploadHandle 图片上传句柄, NET_DVR_PicUpload 的返回值

返回值: -1 表示失败, 0~100 表示图片上传进度。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](#)

6.15.5 获取图片上传的状态 **NET_DVR_GetPicUploadState**

函数: LONG NET_DVR_GetPicUploadState(LONG IUploadHandle)

参数: [in] IUploadHandle 图片上传句柄, NET_DVR_PicUpload 的返回值

返回值: -1 表示失败, 其他值定义如下: 1- 完成, 2- 正在上传, 3- 上传失败, 4- 未知错误。

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](#)

6.15.6 关闭图片上传 **NET_DVR_CloseUploadHandle**

函 数： BOOL NET_DVR_CloseUploadHandle(LONG IUploadHandle)
参 数： [in] IUploadHandle 图片上传句柄，NET_DVR_PicUpload 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： 关闭上传句柄，释放资源。

[返回目录](#)

6.15.7 删除底图 **NET_DVR_RemoteControl**

函 数： BOOL NET_DVR_RemoteControl(LONG IUserID, DWORD dwCommand, LPVOID lpInBuffer, DWORD dwInBufferSize)
参 数： [in] IUserID 登录主控板，NET_DVR_Login_V40 返回值
[in] dwCommand 控制命令，详见表 6.80
[in] lpInBuffer 输入参数，跟控制命令相关，详见表 6.80
[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，如表 6.80 所示。

表 6.80 底图控制命令

dwCommand 宏定义	宏定义值	控制功能	lpInBuffer 对应结构体
NET_DVR_DELETE_PICTURE	9126	删除底图	4 字节底图序号

[返回目录](#)

6.16 LOGO 配置、上传和控制

6.16.1 获取 LOGO 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_V40 的返回值
[in] dwCommand 设备配置命令，详见表 6.81
[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 6.81
[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 6.81 所示。

表 6.81 LOGO 参数获取

dwCommand 宏定义	dwCommand 含义	IChannel	lpOutBuffer 对应结构体	宏定义值
NET_DVR_GET_MATRIX_LOGO_CFG	获取 Logo 参数	Logo 序号	NET_DVR_MATRIX_LOGO_CFG	9025
NET_DVR_GET_WIN_LOGO_CFG	获取窗口 Logo 参数	子窗口号(组合)	NET_DVR_WIN_LOGO_CFG	9027

[返回目录](#)

6.16.2 设置 LOGO 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_V40 的返回值
[in] dwCommand 设备配置命令, 详见表 6.82
[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 6.82
[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 6.82 所示。

表 6.82 LOGO 参数设置

dwCommand 宏定义	dwCommand 含义	IChannel	lpInBuffer 对应结构体	宏定义值
NET_DVR_SET_MATRIX_LOGO_CFG	设置 Logo 参数	Logo 序号	NET_DVR_MATRIX_LOGO_CFG	9026
NET_DVR_SET_WIN_LOGO_CFG	设置窗口 Logo 参数	子窗口号(组合)	NET_DVR_WIN_LOGO_CFG	9028

[返回目录](#)

6.16.3 LOGO 上传 **NET_DVR_UploadLogo_NEW**

函 数: BOOL NET_DVR_UploadLogo_NEW(LONG IUserID, DWORD dwLogoNo, LPNET_DVR_MATRIX_LOGO_INFO lpLogoInfo, char *sLogoBuffer)

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V40 的返回值
[in] dwLogoNo LOGO 序号
[in] lpLogoInfo LOGO 的参数, 详见结构体 [NET_DVR_MATRIX_LOGO_INFO](#)
[in] sLogoBuffer LOGO 数据缓冲区, 最大 100K, 图像的宽和高必须是 32 的倍数

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: Logo 要求 bmp 格式, 24 位图, 支持分辨率: 32*32 ~ 256*128。

[返回目录](#)

6.16.4 LOGO 下载 **NET_DVR_DownloadLogo**

函 数: BOOL NET_DVR_DownloadLogo(LONG IUserID, DWORD dwLogoNo, LPNET_DVR_MATRIX_LOGO_INFO lpLogoInfo, char *sLogoBuffer, DWORD dwLogoBufferSize)

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V40 的返回值
[in] dwLogoNo LOGO 序号
[out] lpLogoInfo LOGO 信息, 主要返回 Logo 图片文件的大小, 详见结构体 [NET_DVR_MATRIX_LOGO_INFO](#)
[out] sLogoBuffer LOGO 数据缓冲区
[in] dwLogoBufferSize 缓冲区大小

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: 设备是否支持 LOGO 下载能力, 可以通过设备能力集进行判断, 对应电视墙能力集 ([WallAbility](#)), 相关接口: [NET_DVR_GetDeviceAbility](#), 能力集类型: WALL_ABILITY, 节点: <DownloadLogo>。

[返回目录](#)

6.16.5 删除 LOGO NET_DVR_RemoteControl

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

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V40 返回值
[in] dwCommand 控制命令, 详见表 6.83
[in] lpInBuffer 输入参数, 跟控制命令相关, 详见表 6.83
[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, 如表 6.83 所示。设备是否支持 LOGO 删除能力, 可以通过设备能力集进行判断, 对应电视墙能力集 ([WallAbility](#)), 相关接口: [NET_DVR_GetDeviceAbility](#), 能力集类型: WALL_ABILITY, 节点: <LogoDelete>。

表 6.83 LOGO 控制命令

dwCommand 宏定义	宏定义值	控制功能	lpInBuffer 对应结构体
NET_DVR_DELETE_LOGO	9029	删除 LOGO	4 字节 LOGO 序号

[返回目录](#)

6.17 解码 OSD 配置

6.17.1 获取解码 OSD 能力集 NET_DVR_GetSTDAbility

函 数: BOOL NET_DVR_GetSTDAbility(LONG IUserID, DWORD dwAbilityType, LPNET_DVR_STD_ABILITY lpAbilityParam)

参 数: [in] IUserID 用户 ID 号, NET_DVR_Login_V40 的返回值
[in] dwAbilityType 能力类型, 具体定义见表 6.84
[in&out] lpAbilityParam 设备能力集参数 (包括输入和输出参数), 不同的能力集对应不同的输入输出参数

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 6.84 获取设备能力集

宏定义	宏定义值	含义	IpCondBuffer	IpOutBuffer
NET_DVR_GET_SUBWND_DECODE_OSD_CAP	9186	获取子窗口解码 OSD 信息能力集	4 字节子窗口号(组合)	DecodeOSD

[返回目录](#)

6.17.2 获取解码 OSD 参数 **NET_DVR_GetSTDConfig**

- 函 数: BOOL NET_DVR_GetSTDConfig(LONG IUserID, DWORD dwCommand, LPNET_DVR_STD_CONFIG lpConfigParam)
- 参 数: [in]IUserID 用户 ID 号, NET_DVR_Login_V40 的返回值
[in]dwCommand 设备配置命令, 详见表 6.85
[in&out]lpConfigParam 配置输入输出参数, 详见结构体: [NET_DVR_STD_CONFIG](#), 不同的配置功能对应不同的输入输出参数, 详见表 6.85
- 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: 获取配置参数时, lpConfigParam 结构体中的 lpInBuffer 无效, 设为 NULL。对于不同的配置功能(dwCommand), lpConfigParam 中 lpCondBuffer、lpOutBuffer 分别对应不同的内容, 具体如表 6.85 所示。

表 6.85 获取设备参数

dwCommand 宏定义	dwCommand 含义	IpCondBuffer	IpOutBuffer	宏定义值
NET_DVR_GET_SUBWND_DECODE_OSD	获取子窗口解码 OSD 信息	4 字节子窗口号(组合)	NET_DVR_SUBWND_DECODE_OSD	9183
NET_DVR_GET_SUBWND_DECODE_OSD_ALL	获取所有子窗口解码 OSD 信息	4 字节窗口号(组合)	NET_DVR_SUBWND_DECODE_OSD_LIST	9184

[返回目录](#)

6.17.3 设置解码 OSD 参数 **NET_DVR_SetSTDConfig**

- 函 数: BOOL NET_DVR_SetSTDConfig(LONG IUserID, DWORD dwCommand, LPNET_DVR_STD_CONFIG lpConfigParam)
- 参 数: [in]IUserID 用户 ID 号, NET_DVR_Login_V40 的返回值
[in]dwCommand 设备配置命令, 详见表 6.86
[in&out]lpConfigParam 配置输入输出参数, 详见结构体: [NET_DVR_STD_CONFIG](#), 不同的配置功能对应不同的输入输出参数, 详见表 6.86
- 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: 设置配置参数时, lpConfigParam 结构体里面的 lpOutBuffer 无效, 设为 NULL。对于不同的配置功能(dwCommand), lpConfigParam 中的 lpCondBuffer、lpInBuffer 分别对应不同的内容, 具体如表 6.86 所示。

表 6.86 设置设备命令

dwCommand 宏定义	dwCommand 含义	IpCondBuffer	IpInBuffer	宏定义值
NET_DVR_SET_SUBWND_DECODE_OSD	设置子窗口解码 OSD 信息	4 字节子窗口号 (组合)	NET_DVR_SUBWND_DECODE_OSD	9185

[返回目录](#)

6.18 图片回显

6.18.1 开始图片回显 **NET_DVR_StartPicPreview**

函 数: LONG NET_DVR_StartPicPreview(LONG lUserID, LPNET_DVR_START_PIC_VIEW_INFO lpStruStartPicView, SCREENPICDATACB PicDataCallback, void *pUserData)

参 数:

- [in] lUserID 登录主控板, NET_DVR_Login_V40 的返回值
- [in] lpStruStartPicView 回显参数, 详见结构体 [NET_DVR_START_PIC_VIEW_INFO](#)
- [in] PicDataCallback 图片回调函数
- [in] pUserData 用户数据

```
typedef void(CALLBACK *SCREENPICDATACB)(LONG nScreenPicHandle, DWORD dwDataType, BYTE *pBuffer, DWORD dwBufSize, void *pUser)
```

[out] nScreenPicHandle 图片回显句柄

[out] dwDataType 数据类型

```
enum _PIC_PREVIEW_DATATYPE_{
    NET_DVR_DATA_BMP    = 0, //BMP 图片数据
    NET_DVR_START_CALLBACK, //起始回调
    NET_DVR_STREAM_SYSHEAD, //系统头数据
    NET_DVR_STREAM_DATA  //流数据
}TRAFFIC_SCENE_MODE
```

[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

返回值: -1 表示失败, 其他值作为 NET_DVR_StopScreenPic 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: 本地信号源调用该回显接口在客户端显示画图, 网络信号源使用网络预览接口。

[返回目录](#)

6.18.2 停止图片回显 **NET_DVR_StopScreenPic**

函 数: BOOL NET_DVR_StopScreenPic(LONG nScreenPicHandle)

参 数: [in] nScreenPicHandle 图片回显句柄, NET_DVR_StartPicPreview 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](#)

6.19 图像微调

6.19.1 图像微调 **NET_DVR_MatrixPicAdjust**

- 函 数: `BOOL NET_DVR_MatrixPicAdjust(LONG IUserID, DWORD dwDispChan, DWORD dwPicAdjust, DWORD dwCmdParam)`
- 参 数:
- | | |
|-------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>[in] IUserID</code> | NET_DVR_Login_V40 的返回值(视频综合平台需要登录编码子系统) |
| <code>[in] dwDispChan</code> | 对于多屏控制器, 该参数表示信号源序号; 对于视频综合平台, 该参数为编码子系统的通道号 |
| <code>[in] dwPicAdjust</code> | 微调命令, 详见表 6.87 |
| <code>[in] dwCmdParam</code> | 图像微调参数, 大于 0 的整型值, 表示调整大小, 单位为: 个(像素)。例如: <code>dwPicAdjust</code> 为 <code>DECODEPIC_LEFTADJUST</code> 时, <code>dwCmdParam</code> 表示图片向左移动的像素点个数; <code>dwPicAdjust</code> 为 <code>DECODEPIC_HEIGHTADJUST</code> 时, <code>dwCmdParam</code> 表示调整后的高度; <code>dwPicAdjust</code> 为 <code>DECODEPIC_WIDTHADJUST</code> 时, <code>dwCmdParam</code> 表示调整后的宽度 |
- 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: 该接口是针对编码子系统的操作。设备是否支持图像微调能力, 可以通过设备能力集进行判断, 对应图像参数能力集 (`VideoPicAbility`), 相关接口: [NET_DVR_GetDeviceAbility](#), 能力集类型: `DEVICE_VIDEOPIC_ABILITY`, 节点: `<PicAdjust>`。

表 6.87 图像微调命令

dwPicAdjust 宏定义	宏定义值	含义
DECODEPIC_LEFTADJUST	1	图像左移
DECODEPIC_RIGHTADJUST	2	图像右移
DECODEPIC_UPADJUST	3	图像上移
DECODEPIC_DOWNADJUST	4	图像下移
DECODEPIC_REDUCEADJUST	5	图像缩小
DECODEPIC_AUTOADJUST	8	图像自动调整
DECODEPIC_HEIGHTADJUST	9	图像高度调整
DECODEPIC_WIDTHADJUST	10	图像宽度调整

[返回目录](#)

6.20 透明通道

6.20.1 建立透明通道 **NET_DVR_SerialStart**

- 函 数: `LONG NET_DVR_SerialStart(LONG IUserID, LONG ISerialPort, fSerialDataCallBack cbSerialDataCallBack, DWORD dwUser)`

参 数: [in]IUserID 登录主控板, NET_DVR_Login_V40 的返回值
[in]ISerialPort 串口号: 1- 232 串口, 2- 485 串口
[in]cbSerialDataCallBack 透明通道数据回调函数
[in]dwUser 用户数据

```
typedef void(CALLBACK *fSerialDataCallBack)(LONG ISerialHandle, char *pRecvDataBuffer, DWORD dwBufSize, DWORD dwUser)
```

[out]ISerialHandle NET_DVR_SerialStart 的返回值
[out]pRecvDataBuffer 存放数据的缓冲区指针
[out]dwBufSize 数据大小
[out]dwUser 用户数据

返回值: -1 表示失败, 其他值作为 NET_DVR_SerialSend 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: 需要从回调函数得到数据解码器必须支持数据回传, 否则发送成功, 回调依然不会有返回。

[返回目录](#)

6.20.2 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 使用 485 串口时有效, 从 1 开始; 232 串口作为透明通道时该值设置为 0
[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](#)

6.20.3 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](#)

6.21 云台控制

6.21.1 云台控制 NET_DVR_PTZControlWithSpeed_Other

函 数: BOOL NET_DVR_PTZControlWithSpeed_Other(LONG IUserID, LONG IChannel, DWORD

dwPTZCommand, DWORD dwStop, DWORD dwSpeed)

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V40 的返回值
 [in] IChannel 对于视频综合平台, 该参数为子窗口号(组合)或者编码通道号:
 (0x40|1 字节设备号)+1 字节子板号+2 字节编码通道号
 对于多屏控制器, 该参数为子窗口号(组合)或者输入源:
 (0x40|1 字节设备号)+1 字节保留+2 字节输入源序号
 [in] dwPTZCommand 云台控制命令, 详见表 6.88
 [in] dwStop 云台停止动作或开始动作: 0- 开始, 1- 停止
 [in] dwSpeed 云台控制的速度, 用户按不同解码器的速度控制值设置, 取值范围[1,7]

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:

- 对云台实施的每一个动作都需要调用该接口两次, 分别是开始和停止控制, 由接口中的最后一个参数 (dwStop) 决定。
- 设备是否支持通过主控板控制云台能力, 可以通过设备能力集进行判断, 对应软硬件能力集 (BasicCapability), 相关接口: [NET_DVR_GetDeviceAbility](#), 能力集类型: DEVICE_SOFTHARDWARE_ABILITY, 节点: <SupportMainBoardCameraPTZ>。
- 通过主控板控制云台之前, 需要调用 [NET_DVR_GetDVRConfig](#)(命令: NET_DVR_GET_CAMERACHAN_SERIALCFG)、[NET_DVR_SetDVRConfig](#)(命令: NET_DVR_SET_CAMERACHAN_SERIALCFG)配置 Camera 通道绑定串口参数。

表 6.88 云台控制命令

宏定义	宏定义值	含义
LIGHT_PWRON	2	接通灯光电源
WIPER_PWRON	3	接通雨刷开关
FAN_PWRON	4	接通风扇开关
HEATER_PWRON	5	接通加热器开关
AUX_PWRON1	6	接通辅助设备开关
AUX_PWRON2	7	接通辅助设备开关
ZOOM_IN	11	焦距变大(倍率变大)
ZOOM_OUT	12	焦距变小(倍率变小)
FOCUS_NEAR	13	焦点前调
FOCUS_FAR	14	焦点后调
IRIS_OPEN	15	光圈扩大
IRIS_CLOSE	16	光圈缩小
TILT_UP	21	云台上仰
TILT_DOWN	22	云台下俯
PAN_LEFT	23	云台左转
PAN_RIGHT	24	云台右转
UP_LEFT	25	云台上仰和左转

UP_RIGHT	26	云台上仰和右转
DOWN_LEFT	27	云台下俯和左转
DOWN_RIGHT	28	云台下俯和右转
PAN_AUTO	29	云台左右自动扫描
TILT_DOWN_ZOOM_IN	58	云台下俯和焦距变大(倍率变大)
TILT_DOWN_ZOOM_OUT	59	云台下俯和焦距变小(倍率变小)
PAN_LEFT_ZOOM_IN	60	云台左转和焦距变大(倍率变大)
PAN_LEFT_ZOOM_OUT	61	云台左转和焦距变小(倍率变小)
PAN_RIGHT_ZOOM_IN	62	云台右转和焦距变大(倍率变大)
PAN_RIGHT_ZOOM_OUT	63	云台右转和焦距变小(倍率变小)
UP_LEFT_ZOOM_IN	64	云台上仰和左转和焦距变大(倍率变大)
UP_LEFT_ZOOM_OUT	65	云台上仰和左转和焦距变小(倍率变小)
UP_RIGHT_ZOOM_IN	66	云台上仰和右转和焦距变大(倍率变大)
UP_RIGHT_ZOOM_OUT	67	云台上仰和右转和焦距变小(倍率变小)
DOWN_LEFT_ZOOM_IN	68	云台下俯和左转和焦距变大(倍率变大)
DOWN_LEFT_ZOOM_OUT	69	云台下俯和左转和焦距变小(倍率变小)
DOWN_RIGHT_ZOOM_IN	70	云台下俯和右转和焦距变大(倍率变大)
DOWN_RIGHT_ZOOM_OUT	71	云台下俯和右转和焦距变小(倍率变小)
TILT_UP_ZOOM_IN	72	云台上仰和焦距变大(倍率变大)
TILT_UP_ZOOM_OUT	73	云台上仰和焦距变小(倍率变小)

[返回目录](#)

6.21.2 3 维带速度云台控制 **NET_DVR_RemoteControl**

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

参 数: [in] lUserID 登录主控板, NET_DVR_Login_V40 返回值
[in] dwCommand 控制命令, 详见表 6.89
[in] lpInBuffer 输入参数, 跟控制命令相关, 详见表 6.89
[in] dwInBufferSize 输入参数长度

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: 设备是否支持 3 维带速度云台控制能力, 可以通过设备能力集进行判断, 对应软硬件能力集 (BasicCapability), 相关接口: [NET_DVR_GetDeviceAbility](#), 能力集类型: DEVICE_SOFTWARE_ABILITY, 节点: <Support3DPTZ>。

表 6.89 远程控制命令

dwCommand 宏定义	宏定义值	控制功能	lpInBuffer 对应结构体
NET_DVR_PTZ_3D_SPEED	1765	3 维带速度的云台控制	NET_DVR_PTZ_3D_SPEED_CONTROL

[返回目录](#)

6.21.3 获取设备支持的云台协议列表 **NET_DVR_GetPTZProtocol_Ex**

函 数： BOOL NET_DVR_GetPTZProtocol_Ex(LONG IUserID, LONG IChannel, NET_DVR_PTZCFG *pPtzcfg)

参 数： [in] IUserID 登录主控板，NET_DVR_Login_V40 的返回值
 [in] IChannel 通道号
 [out] lpProtoList 云台协议列表，详见结构体 [NET_DVR_PTZCFG](#)

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： 该接口用于获取当前登录对应的设备所支持的云台协议。

[返回目录](#)

6.21.4 获取云台控制参数 **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 X86 子板，NET_DVR_Login_V40 的返回值
 [in] dwCommand 设备配置命令，详见表 6.90
 [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 6.90
 [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 6.90 所示。设备是否支持图像融合能力或者支持的融合通道个数，可以通过设备能力集进行判断，对应相机拼接能力集 ([CamFusionAbility](#))，相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：CAM_FUSION_ABILITY。

表 6.90 获取云台参数

dwCommand 宏定义	dwCommand 含义	IChannel	lpOutBuffer 对应结构体	宏定义值
NET_DVR_PTZCFG_INPUTSTREAM_GET	获取输入源反向云台控制配置	信号源序号	NET_DVR_INPUTSTREAM_PTZCFG_PARAM	1777

[返回目录](#)

6.21.5 设置云台控制参数 **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 X86 子板，NET_DVR_Login_V40 的返回值

[in] dwCommand	设备配置命令，详见表 6.91
[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 6.91
[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 6.91 所示。设备是否支持图像融合能力或者支持的融合通道个数，可以通过设备能力集进行判断，对应相机拼接能力集（[CamFusionAbility](#)），相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：CAM_FUSION_ABILITY。

表 6.91 设置云台参数

dwCommand 宏定义	dwCommand 含义	IChannel	lpInBuffer 对应结构体	宏定义值
NET_DVR_PTZCFG_INPUTSTREAM_SET	设置输入源反向云台控制配置	信号源序号	NET_DVR_INPUTSTREAM_PTZCFG_PARAM	1778

[返回目录](#)

6.22 相机拼接配置

6.22.1 获取流来源信息 [NET_DVR_GetDeviceConfig](#)

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

参 数：

[in] IUserID	登录 X86 子板，NET_DVR_Login_V40 的返回值
[in] dwCommand	设备配置命令，详见表 6.92
[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 6.93
[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	设备备返回的参数内容（详见表 6.93），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 6.92 批量参数获取命令

dwCommand 宏定义	dwCommand 含义	宏定义值
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。

因。

说明： 该接口是带有发送数据的批量获取监控点配置信息的通用接口。IpInBuffer 指定需要获取的 dwCount 个配置信息，IpOutBuffer 保存获取到的 dwCount 个配置信息。
全部获取时 dwCount 置为 0xffffffff，IpInBuffer 置为 NULL，dwInBufferSize 置为 0，IpStatusList 置为 NULL，IpOutBuffer 前 4 个字节为返回的个数。如果设置的 IpOutBuffer 缓冲区不足，仅返回部分信息，可以根据返回的个数（前 4 字节的值）重新获取。

表 6.93 批量参数获取结构

dwCommand 宏定义	IpInBuffer 对应结构体	IpOutBuffer 对应结构体
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

[返回目录](#)

6.22.2 设置流来源信息 **NET_DVR_SetDeviceConfig**

函数： BOOL NET_DVR_SetDeviceConfig(LONG IUserID,DWORD dwCommand,DWORD dwCount,LPVOID IpInBuffer,DWORD dwInBufferSize,LPVOID IpStatusList, LPVOID IpInParamBuffer,DWORD dwInParamBufferSize)

参数：

- [in] IUserID 登录 X86 子板，NET_DVR_Login_V40 的返回值
- [in] dwCommand 设备配置命令，详见表 6.94
- [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] IpInBuffer 配置条件缓冲区，详见表 6.95
- [in] dwInBufferSize 缓冲区长度
- [out] IpStatusList 错误信息列表，和要查询的监控点一一对应，例如 IpStatusList[2] 就对应 IpInBuffer[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
- [in] IpInParamBuffer 需要设置给设备的参数内容（详见表 6.95），和要查询的监控点一一对应。如果某个监控点对应的 IpStatusList 信息为大于 0 值，表示对应的 IpInBuffer 设置失败，为 0 则设置成功
- [in] dwInParamBufferSize 设置内容缓冲区大小

表 6.94 批量参数配置命令

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

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 IpStatusList[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。IpInBuffer 指定需要设置的 dwCount 个监控点信息，IpOutBuffer 保存将要设置的 dwCount 个监控点的配置信息。
不同的获取功能对应不同的结构体和命令号，如表 6.95 所示。

表 6.95 批量参数配置结构

dwCommand 宏定义	IpInBuffer 对应结构体	IpInParamBuffer 对应结构体
NET_DVR_SET_STREAM_SRC_INFO	dwCount 个 NET_DVR_STREAM_INFO	dwCount 个 NET_DVR_STREAM_SRC_INFO

[返回目录](#)

6.22.3 获取相机拼接参数 **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 X86 子板, NET_DVR_Login_V40 的返回值
[in] dwCommand 设备配置命令, 详见表 6.96
[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 6.96
[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 6.96 所示。设备是否支持图像融合能力或者支持的融合通道个数, 可以通过设备能力集进行判断, 对应相机拼接能力集 ([CamFusionAbility](#)), 相关接口: [NET_DVR_GetDeviceAbility](#), 能力集类型: CAM_FUSION_ABILITY。

表 6.96 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_FUSION_SCALE	获取图像融合规模	融合通道号	NET_DVR_FUSION_SCALE	9151
NET_DVR_GET_FUSION_CFG	获取图像融合参数	融合通道号	NET_DVR_FUSION_CFG	9053

[返回目录](#)

6.22.4 设置相机拼接参数 **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 X86 子板, NET_DVR_Login_V40 的返回值
[in] dwCommand 设备配置命令, 详见表 6.97
[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 6.97
[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 6.97 所示。设备是否支持图像融合能力或者支持的融合通道个数, 可以通过设备能力集进行判断, 对应相机拼接能力集 ([CamFusionAbility](#)), 相关接口: [NET_DVR_GetDeviceAbility](#), 能力集类型: CAM_FUSION_ABILITY。

表 6.97 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_FUSION_SCALE	设置图像融合规模	融合通道号	NET_DVR_FUSION_SCALE	9150
NET_DVR_SET_FUSION_CFG	设置图像融合参数	融合通道号	NET_DVR_FUSION_CFG	9052

[返回目录](#)

6.23 信号源拼接裁剪配置

6.23.1 获取拼接裁剪参数 **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_V40 的返回值
- [in] dwCommand 设备配置命令，详见表 6.98
- [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 6.98
- [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 6.98 所示。

表 6.98 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_SIGNAL_JOINT	获取信号源绑定配置 (信号源拼接)	4 字节拼接序号	NET_DVR_SIGNAL_JOINT_CFG	1766
NET_DVR_SIGNAL_CUTPARAM_GET	获取信号源裁剪参数	4 字节信号源序号	NET_DVR_SIGNAL_CUTPARAM	1779

[返回目录](#)

6.23.2 批量获取拼接参数 **NET_DVR_GetDeviceConfig**

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

参 数：

- [in] IUserID 登录主控板，NET_DVR_Login_V40 的返回值
- [in] dwCommand 设备配置命令，详见表 6.99
- [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 6.100
- [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 6.100），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 6.99 批量参数获取命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_GET_ALL_SIGNAL_JOINT	获取所有信号源绑定配置(dwCount 为 0)	1772

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 IpStatusList[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监控点配置信息的通用接口。IpInBuffer 指定需要获取的 dwCount 个配置信息, IpOutBuffer 保存获取到的 dwCount 个配置信息。
全部获取时 dwCount 置为 0xffffffff, IpInBuffer 置为 NULL, dwInBufferSize 置为 0, IpStatusList 置为 NULL, IpOutBuffer 前 4 个字节为返回的个数。如果设置的 IpOutBuffer 缓冲区不足, 仅返回部分信息, 可以根据返回的个数(前 4 字节的值)重新获取。

表 6.100 批量参数获取结构

dwCommand 宏定义	IpInBuffer 对应结构体	IpOutBuffer 对应结构体
NET_DVR_GET_ALL_SIGNAL_JOINT	NULL	4 字节 N + N 个 NET_DVR_SIGNAL_JOINT_CFG

[返回目录](#)

6.23.3 批量设置拼接参数 **NET_DVR_SetDeviceConfigEx**

函数: BOOL NET_DVR_SetDeviceConfigEx(LONG iUserID, DWORD dwCommand, DWORD dwCount, NET_DVR_IN_PARAM *IpInParam, NET_DVR_OUT_PARAM *IpOutParam)

参数: [in] iUserID 登录主控板, NET_DVR_Login_V40 的返回值
[in] dwCommand 设备配置命令, 详见表 6.101
[in] dwCount 批量配置个数, dwCount 为 1
[in] IpInParam 输入参数缓冲区, 详见结构体: [NET_DVR_IN_PARAM](#), 其中参数取值根据不同的 dwCommand 而不同, 详见表 6.102
[in] IpOutParam 输出参数缓冲区, 详见结构体: [NET_DVR_OUT_PARAM](#), 其中参数取值根据不同的 dwCommand 而不同, 详见表 6.102

表 6.101 批量配置命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_SET_SIGNAL_JOINT	设置信号源拼接绑定配置	1767

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 IpStatusList[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。此接口实现信号源拼接功能, 信号源拼接是将超高清信号分割成 N 等份, 通过拼接板的 N 个输入口进行输入, 多个输入组合成一个信号源, 此信号源可以像普通前端源一样进行上墙。调用设置接口之前, 可以调用 [NET_DVR_GetDVRConfig](#)(命令: NET_DVR_GET_SIGNAL_JOINT) 获取设备当前的信号源拼接绑定配置参数。

表 6.102 批量参数配置结构

dwCommand 宏定义	取值含义
NET_DVR_SET_SIGNAL_JOINT	设置信号源拼接绑定配置，dwCount 为 1
输入输出缓冲区参数	参数取值
lpInParam->struCondBuf.pBuf	输入条件缓冲区，dwCount 个 4 字节拼接序号
lpInParam->struCondBuf.nLen	输入条件缓冲区长度
lpInParam->struInParamBuf.pBuf	输入参数缓冲区，dwCount 个 NET_DVR_SIGNAL_JOINT_CFG
lpInParam->struInParamBuf.nLen	输入参数缓冲区长度
lpInParam->dwRecvTimeout	数据接收超时时间，填 0 采用接口默认超时
lpOutParam->struOutBuf.pBuf	输出参数缓冲区，dwCount 个 4 字节拼接信号源序号
lpOutParam->struOutBuf.nLen	输出参数缓冲区长度
lpOutParam->lpStatusList	状态缓冲区，对应 dwCount 个参数配置的结果，例如 lpStatusList[2]就对应 lpInParam->struInParamBuf.pBuf[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

[返回目录](#)

6.23.4 拼接裁剪控制 **NET_DVR_RemoteControl**

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

参 数： [in] IUserID 登录主控板，NET_DVR_Login_V40 返回值
[in] dwCommand 控制命令，详见表 6.103
[in] lpInBuffer 输入参数，跟控制命令相关，详见列表
[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，如表 6.103 所示。

表 6.103 远程控制命令

dwCommand 宏定义	宏定义值	控制功能	lpInBuffer 对应结构体
NET_DVR_SIGNAL_CUT	1768	信号源裁剪	NET_DVR_SIGNAL_CUTPARAM
NET_DVR_DEL_SIGNAL_JOINT	1782	删除拼接信号源	4 字节拼接序号

[返回目录](#)

6.24 光端机相关配置

6.24.1 获取光信号相关参数 **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 X86 子板，NET_DVR_Login_V40 的返回值
[in] dwCommand 设备配置命令，详见表 6.104

[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 6.104
[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 6.104 所示。

表 6.104 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_NS_RING_CFG	获取光纤板环网配置	4 字节环网号(组合)	NET_DVR_NS_RING_CFG	9212
NET_DVR_GET_NS_RING_STATUS	获取光纤板环网状态	4 字节环网号(组合)	NET_DVR_NS_RING_STATUS	9214
NET_DVR_GET_OPTICAL_PORT_INFO	获取光口信息	光口号(组合)	NET_DVR_OPTICAL_PORT_INFO	9220
NET_DVR_GET_OPTICAL_CHAN_RELATE_CFG	获取编码通道关联光口输入源参数	编码通道号(组合)	NET_DVR_OPTICAL_CHAN_RELATE_CFG	9222

[返回目录](#)

6.24.2 设置光信号相关参数 **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 X86 子板，NET_DVR_Login_V40 的返回值
[in] dwCommand 设备配置命令，详见表 6.105
[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 6.105
[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 6.105 所示。

表 6.105 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_NS_RING_CFG	设置光纤板环网配置	4 字节环网号(组合)	NET_DVR_NS_RING_CFG	9213
NET_DVR_SET_OPTICAL_PORT_INFO	设置光口信息	光口号(组合)	NET_DVR_OPTICAL_PORT_INFO	9220
NET_DVR_SET_OPTICAL_CHAN_RELATE_CFG	设置编码通道关联光口输入源参数	编码通道号(组合)	NET_DVR_OPTICAL_CHAN_RELATE_CFG	9223

[返回目录](#)

6.25 获取 IPC 协议列表

6.25.1 获取设备支持的 IPC 协议列表 **NET_DVR_GetIPCProtoList_V41**

函 数： NET_DVR_GetIPCProtoList_V41(LONG IUserID, LPNET_DVR_IPC_PROTO_LIST_V41 lpProtoList)

参 数： [in]IUserID 登录主控板，NET_DVR_Login_V40 的返回值
[out]lpProtoList IPC 协议列表，详见结构体 [NET_DVR_IPC_PROTO_LIST_V41](#)

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： 该接口用于获取当前设备所支持的外接 IPC 的协议。调用该接口时，先将 lpProtoList ->pBuffer 置为 NULL、lpProtoList ->dwBufLen 置为 0，获取到有效 IPC 协议数目；然后根据数目分配缓冲区，再调用一次接口，获取具体协议列表信息。

[返回目录](#)

6.26 获取设备状态

6.26.1 获取设备状态信息 **NET_DVR_GetDeviceStatus**

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

参 数： [in] IUserID 登录主控板，NET_DVR_Login_V40 返回值
[in] dwCommand 设备配置命令，详见表 6.106
[in] dwCount 批量获取个数个数，0 和 1 都表示获取 1 个设备信息
[in] lpInBuffer 配置条件缓冲区，详见表 6.106。只获取一个设备的状态时，该参数可设为 NULL
[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 (1 个 32 位无符号整数值)，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 6.106），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。如果只获取一个设备的状态且 lpInBuffer 设为 NULL 时，该参数可设为 NULL
[in] dwOutBufferSize 输出缓冲区大小

返回值： TRUE 表示成功，但不代表每一个状态获取都成功，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设备状态信息的通用接口。不同的获取功能对应不同的结构体和命令号，如表 6.106 所示。

表 6.106 解码状态信息

dwCommand 宏定义	dwCommand 含义	lpInBuffer 对应结构体	lpOutBuffer 对应结构体	宏定义值
NET_DVR_GET_DEVICE_RUN_STATUS	获取设备运行状态	NULL	dwCount 个 NET_DVR_DEVICE_RUN_STATUS	1618

[返回目录](#)

6.26.2 设备在线状态检测 **NET_DVR_RemoteControl**

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

参 数: [in] IUserID 登录主控板, NET_DVR_Login_V40 返回值
[in] dwCommand 控制命令, 详见表 6.107
[in] lpInBuffer 输入参数, 跟控制命令相关, 详见表 6.107
[in] dwInBufferSize 输入参数长度

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: 该控制命令用于手动检测设备是否在线, 接口返回 TRUE 表示在线, FALSE 表示与设备通信失败或者返回错误状态。设备在线状态自动巡检功能通过 [NET_DVR_SetSDKLocalCfg](#) (配置类型: NET_SDK_LOCAL_CFG_TYPE_CHECK_DEV) 进行配置。

表 6.107 设备在线状态检测

dwCommand 宏定义	宏定义值	控制功能	lpInBuffer 对应结构体
NET_DVR_CHECK_USER_STATUS	20005	检测设备是否在线	NULL

[返回目录](#)

6.26.3 获取视频综合平台状态 **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_V40 的返回值
[in] dwCommand 设备配置命令, 详见表 6.108
[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 6.108
[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 6.108 所示。

表 6.108 获取设备状态

dwCommand 宏定义	dwCommand 含义	IChannel	lpOutBuffer 对应结构体	宏定义值
NET_DVR_GET_MATRIX_STATUS	获取视频综合平台状态	NULL	NET_DVR_MATRIX_STATUS	1762

[返回目录](#)

6.27 EDID 文件管理

EDID 文件上传

6.27.1 上传文件 **NET_DVR_UploadFile_V40**

函 数： LONG NET_DVR_UploadFile_V40(LONG IUserID, DWORD dwUploadType, LPVOID lpInBuffer, DWORD dwInBufferSize, char *sFileName, LPVOID lpOutBuffer, DWORD dwOutBufferSize)

数：

- [in] IUserID 登录主控板，NET_DVR_Login_V40 的返回值
- [in] dwUploadType 上传文件类型，详见表 6.109
- [in] lpInBuffer 不同的 dwUploadType，输入参数不同，详见表 6.109
- [in] dwInBufferSize 输入缓冲区大小
- [in] sFileName 上传文件的绝对路径（包括文件名），最多 128 字节，文件名最多 32 字节（含最后的'\0'）
- [out] lpOutBuffer 输出参数。不同的 dwUploadType，输出参数不同，详见表 6.109
- [in] dwOutBufferSize 输出缓冲区大小

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 6.109 上传文件类型

dwUploadType 宏定义	宏定义值	dwUploadType 含义	lpInBuffer 对应结构体	lpOutBuffer 对应结构体
UPLOAD_EDID_CFG_FILE	26	上传 EDID 配置文件	NULL	4 字节的 EDID 编号

[返回目录](#)

6.27.2 获取文件上传的进度和状态 **NET_DVR_GetUploadState**

函 数： LONG NET_DVR_GetUploadState(LONG IUploadHandle, LPDWORD pProgress)

参 数：

- [in] IUploadHandle 文件上传的句柄，NET_DVR_UploadFile_V40 的返回值
- [out] pProgress 返回的进度值，取值范围：0~100

返回值： -1 表示函数调用失败，其他为上传的状态值：1- 上传成功；2- 正在上传；3- 上传失败；4- 网络断开，状态未知。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](#)

6.27.3 停止文件上传 **NET_DVR_UploadClose**

函 数： BOOL NET_DVR_UploadClose(LONG IUploadHandle)

参 数：

- [in] IUploadHandle 文件上传的句柄，NET_DVR_UploadFile_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](#)

EDID 文件信息配置

6.27.4 获取 EDID 文件信息配置能力集 **NET_DVR_GetSTDAbility**

函 数： BOOL NET_DVR_GetSTDAbility(LONG lUserID, DWORD dwAbilityType, LPNET_DVR_STD_ABILITY lpAbilityParam)

参 数： [in]lUserID 用户 ID 号，NET_DVR_Login_V40 的返回值
[in]dwAbilityType 能力类型，具体定义见表 6.110
[in&out]lpAbilityParam 设备能力集参数（包括输入和输出参数），不同的能力集对应不同的输入输出参数，详见表 6.110

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 6.110 获取设备能力集

宏定义	宏定义值	含义	lpCondBuffer	lpOutBuffer
NET_DVR_GET_EDID_CFG_FILE_INFO_CAP	9182	获取 EDID 配置文件信息能力集	NULL	EdidFileList

[返回目录](#)

6.27.5 获取 EDID 文件信息 **NET_DVR_GetSTDConfig**

函 数： BOOL NET_DVR_GetSTDConfig(LONG lUserID, DWORD dwCommand, LPNET_DVR_STD_CONFIG lpConfigParam)

参 数： [in]lUserID 用户 ID 号，NET_DVR_Login_V40 的返回值
[in]dwCommand 设备配置命令，详见表 6.111
[in&out]lpConfigParam 配置输入输出参数，详见结构体：[NET_DVR_STD_CONFIG](#)，不同的配置功能对应不同的输入输出参数，详见表 6.111

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： 获取配置参数时，lpConfigParam 结构体中的 lpInBuffer 无效，设为 NULL。对于不同的配置功能（dwCommand），lpConfigParam 中 lpCondBuffer、lpOutBuffer 分别对应不同的内容，具体如表 6.111 所示。

表 6.111 获取设备参数

dwCommand 宏定义	dwCommand 含义	lpCondBuffer	lpOutBuffer	宏定义值
NET_DVR_GET_EDID_CFG_FILE_INFO	获取 EDID 配置文件信息	4 字节 EDID 文件编号	NET_DVR_EDID_FILE_INFO	9178
NET_DVR_GET_EDID_CFG_FILE_INFO_LIST	获取所有 EDID 配置文件信息	NULL	NET_DVR_EDID_FILE_INFO_LIST	9179

[返回目录](#)

6.27.6 设置 EDID 文件信息 **NET_DVR_SetSTDConfig**

函 数： BOOL NET_DVR_SetSTDConfig(LONG lUserID, DWORD dwCommand, LPNET_DVR_STD_CONFIG

IpConfigParam)

参 数: [in]IUserID 用户 ID 号, NET_DVR_Login_V40 的返回值
[in]dwCommand 设备配置命令, 详见表 6.112
[in&out]IpConfigParam 配置输入输出参数, 详见结构体: [NET_DVR_STD_CONFIG](#), 不同的配置功能对应不同的输入输出参数, 详见表 6.112

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: 设置配置参数时, IpConfigParam 结构体里面的 IpOutBuffer 无效, 设为 NULL。对于不同的配置功能 (dwCommand), IpConfigParam 中的 IpCondBuffer、IpInBuffer 分别对应不同的内容, 具体如表 6.112 所示。

表 6.112 设置设备命令

dwCommand 宏定义	dwCommand 含义	IpCondBuffer	IpInBuffer	宏定义值
NET_DVR_SET_EDID_CFG_FILE_INFO	设置 EDID 配置文件信息	4 字节 EDID 文件编号	NET_DVR_EDID_FILE_INFO	9180

[返回目录](#)

6.27.7 删除 EDID 文件信息 **NET_DVR_STDControl**

函 数: BOOL NET_DVR_STDControl(LONG IUserID, DWORD dwCommand, LPNET_DVR_STD_CONTROL IpControlParam)

参 数: [in]IUserID 用户 ID 号, NET_DVR_Login_V40 的返回值
[in]dwCommand 控制命令, 详见表 6.113
[in&out]IpControlParam 远程控制输入输出参数, 详见结构体 [NET_DVR_STD_CONTROL](#)

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: 对于不同的配置功能 (dwCommand), IpControlParam 中的 IpCondBuffer 对应不同的内容, 详见表 6.113。

表 6.113 控制命令

dwCommand 宏定义	宏定义值	dwCommand 含义	IpCondBuffer
NET_DVR_DEL_EDID_CFG_FILE_INFO	9181	删除 EDID 配置文件信息 (包括文件)	4 字节 EDID 文件编号

[返回目录](#)

6.28 设备维护

远程升级

6.28.1 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](#)

6.28.2 远程升级 [NET_DVR_Upgrade](#)

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

参 数: [in]IUserID 登录主控板, NET_DVR_Login_V40 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: 对于分布式多屏控制器, 调用该接口仅上传升级包文件, 之后还需要调用 [NET_DVR_StartUpgrade](#) 启动升级。

[返回目录](#)

6.28.3 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](#)

6.28.4 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](#)

6.28.5 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 6.114 所示。

表 6.114 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](#)

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

函 数: BOOL NET_DVR_CloseUpgradeHandle(LONG IUpgradeHandle)
参 数: [in]IUpgradeHandle 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](#)

6.28.7 开始升级子设备 NET_DVR_StartUpgrade

函 数: LONG NET_DVR_StartUpgrade(LONG IUserID, LPNET_DVR_OPERATE_DEVICE lpStruOperate, DVCS_UPGRADESTATE_CB UpgradeStateCallback, void *pUserData)
参 数: [in] IUserID 登录主控板, NET_DVR_Login_V40 返回值
[in] lpStruOperate 设备升级操作参数, 详见结构体: NET_DVR_OPERATE_DEVICE
[in] UpgradeStateCallback 设备升级回调函数
[in] pUserData 用户数据

```
typedef void(CALLBACK *DVCS_UPGRADESTATE_CB)( LONG nUpgradeHandle, BYTE *pBuffer, DWORD dwBufSize, void *pUser)
```

[out] nUpgradeHandle 当前的升级句柄, NET_DVR_StartUpgrade 的返回值
[out] pBuffer 存放数据的缓冲区指针, 对应结构体 NET_DVR_UPGRADE_STATE
[out] dwBufSize 缓冲区大小
[out] pUser 用户数据
返回值: -1 表示失败, 其他值作为 NET_DVR_StopUpgrade 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: 该接口用于升级分布式多屏控制器的子设备。调用该接口前需要先调用 [NET_DVR_Upgrade](#) 上传升级包数据。

[返回目录](#)

6.28.8 停止升级子设备 **NET_DVR_StopUpgrade**

函 数： LONG NET_DVR_StopUpgrade(LONG IUserID, LPNET_DVR_OPERATE_DEVICE lpStruOperate, LONG IUpgradeHandle)

参 数： [in] IUserID 登录主控板，NET_DVR_Login_V40 返回值
[in] lpStruOperate 设备升级操作参数，详见结构体：NET_DVR_OPERATE_DEVICE，dwSubCommand 赋值为 2
[in] IUpgradeHandle 设备升级句柄，NET_DVR_StartUpgrade 的返回值

返回值： 1 表示成功，0 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： 该接口用于停止升级多屏控制器的子设备。

[返回目录](#)

日志查找

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

函 数： LONG NET_DVR_FindDVRLog_V30(LONG IUserID, LONG ISelectMode, DWORD dwMajorType, DWORD dwMinorType, LPNET_DVR_TIME lpStartTime, LPNET_DVR_TIME lpStopTime, BOOL bOnlySmart = FALSE)

参 数： [in] IUserID 登录主控板，NET_DVR_Login_V40 的返回值
[in] ISelectMode 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
[in] lpStopTime 文件结束时间
[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](#)

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

函 数： LONG NET_DVR_FindNextLog_V30(LONG ILogHandle, LPNET_DVR_LOG_V30 lpLogData)

参 数： [in] ILogHandle 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，详见结构体 [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 6.115。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 6.115 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](#)

6.28.11 查找视频综合平台的日志信息 **NET_DVR_FindDVRLog_Matrix**

函数： LONG NET_DVR_FindDVRLog_Matrix(LONG IUserID, LONG ISelectMode, DWORD dwMajorType, DWORD dwMinorType, LPNET_DVR_VEDIOPLATLOG lpVedioPlatLog, LPNET_DVR_TIME lpStartTime, LPNET_DVR_TIME lpStopTime)

参数：

- [in] IUserID 登录主控板，NET_DVR_Login_V40 的返回值
- [in] ISelectMode 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] lpVedioPlatLog 日志搜索条件，详见结构体 [NET_DVR_VEDIOPLATLOG](#)
- [in] lpStartTime 文件的开始时间，详见结构体 [NET_DVR_TIME](#)
- [in] lpStopTime 文件结束时间，详见结构体 [NET_DVR_TIME](#)

返回值： -1 表示失败，其他值作为 NET_DVR_FindNextLog_MATRIX 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。

[返回目录](#)

6.28.12 逐条获取查找到的平台日志信息 **NET_DVR_FindNextLog_MATRIX**

函数： LONG NET_DVR_FindNextLog_MATRIX (LONG ILogHandle, LPNET_DVR_LOG_MATRIX lpLogData)

参数：

- [in] ILogHandle 日志查找句柄，NET_DVR_FindDVRLog_Matrix 的返回值
- [out] lpLogData 保存日志信息的指针，详见结构体 [NET_DVR_LOG_MATRIX](#)

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 6.116。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 6.116 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： 调用该接口获取查找日志之前，必须先调用 [NET_DVR_FindDVRLog_Matrix](#) 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](#)

6.28.13 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_Matrix 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

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

函数： BOOL NET_DVR_RestoreConfig(LONG lUserID)

参数： [in] lUserID 登录主控板，NET_DVR_Login_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](#)

6.28.15 重启设备 **NET_DVR_RebootDVR**

函数： BOOL NET_DVR_RebootDVR(LONG lUserID)

参数： [in] lUserID 登录主控板，NET_DVR_Login_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](#)

6.28.16 恢复默认参数、重启 **NET_DVR_SetDeviceConfig**

函数： BOOL NET_DVR_SetDeviceConfig(LONG lUserID, DWORD dwCommand, DWORD dwCount, LPVOID lpInBuffer, DWORD dwInBufferSize, LPVOID lpStatusList, LPVOID lpInParamBuffer, DWORD

dwInParamBufferSize)		
参 数:	[in] IUserID	登录主控板, NET_DVR_Login_V40 的返回值
	[in] dwCommand	设备配置命令, 详见表 6.117
	[in] dwCount	要设置的子设备个数
	[in] lpInBuffer	配置条件缓冲区, 详见表 6.118
	[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
	[in] lpInParamBuffer	需要设置给设备的参数内容 (详见表 6.118), 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, 表示对应的 lpInBuffer 设置失败, 为 0 则设置成功
	[in] dwInParamBufferSize	设置内容缓冲区大小

表 6.117 批量参数配置命令

dwCommand 宏定义	dwCommand 含义	宏定义值
NET_DVR_RESTORE_V40	恢复默认参数	1657
NET_DVR_REBOOT_V40	重启	1659

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。lpInBuffer 指定需要设置的 dwCount 个监控点信息, lpOutBuffer 保存将要设置的 dwCount 个监控点的配置信息。不同的获取功能对应不同的结构体和命令号, 如表 6.118 所示。

表 6.118 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_RESTORE_V40	dwCount 个 NET_DVR_DEVICEID_INFO	NULL
NET_DVR_REBOOT_V40	dwCount 个 NET_DVR_DEVICEID_INFO	NULL

该设置命令用于分布式多屏控制器控制其子设备。

[返回目录](#)

配置文件导入导出方式一

- 该方式为阻塞模式, 接口需要等待数据发送或者接收完成、设备返回成功, 等待超时为 30 秒。
- [NET_DVR_MatrixGetConfigFile](#)、[NET_DVR_MatrixSetConfigFile](#) 为视频综合平台的导出导入接口, 其他设备通过 [NET_DVR_GetConfigFile](#)、[NET_DVR_SetConfigFile](#) 实现。

6.28.17 导出配置文件 **NET_DVR_GetConfigFile**

函 数:	BOOL NET_DVR_GetConfigFile(LONG IUserID, char *sFileName)
参 数:	[in]IUserID 登录主控板, NET_DVR_Login_V40 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](#)

6.28.18 导入配置文件 **NET_DVR_SetConfigFile**

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

参 数： [in] lUserID 登录主控板，NET_DVR_Login_V40 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](#)

6.28.19 导出视频综合平台配置文件 **NET_DVR_MatrixGetConfigFile**

函 数： BOOL NET_DVR_MatrixGetConfigFile(LONG lUserID, LPNET_DVR_MATRIXDATABASE lpInter, char *sOutBuffer, DWORD dwOutSize, DWORD *pReturnSize)

参 数： [in] lUserID 登录主控板，NET_DVR_Login_V40 的返回值
[in] lpInter 要导出的数据库参数，详见结构体 [NET_DVR_MATRIXDATABASE](#)
[out] sOutBuffer 存放配置参数的缓冲区
[in] dwOutSize 缓冲区大小
[out] pReturnSize 实际获得的缓冲区大小

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： 视频综合平台支持。

当 sOutBuffer = NULL、dwOutSize = 0 且 pReturnSize != NULL 时用于获取参数配置文件的所需的缓冲区长度；

当 sOutBuffer != NULL 且 dwOutSize != 0 时用于获取参数配置文件的所需的缓冲区内容。

[返回目录](#)

6.28.20 导入视频综合平台配置文件 **NET_DVR_MatrixSetConfigFile**

函 数： BOOL NET_DVR_MatrixSetConfigFile(LONG lUserID, LPNET_DVR_MATRIXDATABASE lpInter, char *sInBuffer, DWORD dwInSize)

参 数： [in] lUserID 登录主控板，NET_DVR_Login_V40 的返回值
[in] lpInter 需要导入的数据库参数，详见结构体 [NET_DVR_MATRIXDATABASE](#)
存放配置参数的缓冲区
[in] sInBuffer 缓冲区大小
[in] dwInSize

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： 视频综合平台支持。

[返回目录](#)

配置文件导入导出方式二

- 设备是否支持该方式导入导出配置文件，通过 [NET_DVR_GetDeviceAbility](#) 获取设备软硬件能力集（DEVICE_SOFTHARDWARE_ABILITY）中的<LongLinkConfigurationFile>字段进行判断。
- 该方式通过 [NET_DVR_UploadFile](#)、[NET_DVR_StartDownload](#) 导入导出配置文件，非阻塞模式，直接返回，客户端通过调用获取状态的接口（[NET_DVR_GetUploadState](#)、[NET_DVR_GetDownloadState](#)）不断获取状态来判断是否成功。

6.28.21 上传文件 **NET_DVR_UploadFile_V40**

函 数： LONG NET_DVR_UploadFile_V40(LONG IUserID, DWORD dwUploadType, LPVOID lpInBuffer, DWORD dwInBufferSize, char *sFileName, LPVOID lpOutBuffer, DWORD dwOutBufferSize)

数： [in] IUserID 登录主控板，NET_DVR_Login_V40 的返回值
[in] dwUploadType 上传文件类型，详见表 6.119
[in] lpInBuffer 不同的 dwUploadType，输入参数不同，详见表 6.119
[in] dwInBufferSize 输入缓冲区大小
[in] sFileName 上传文件的绝对路径（包括文件名），最多 128 字节，文件名最多 32 字节（含最后的'\0'）
[out] lpOutBuffer 输出参数。不同的 dwUploadType，输出参数不同，详见表 6.119
[in] dwOutBufferSize 输出缓冲区大小

表 6.119 上传文件类型

dwUploadType 宏定义	宏定义值	dwUploadType 含义	lpInBuffer 对应结构体	lpOutBuffer 对应结构体
CONFIGURATION_FILE	3	上传配置文件	NULL	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：

[返回目录](#)

6.28.22 获取文件上传的进度和状态 **NET_DVR_GetUploadState**

函 数： LONG NET_DVR_GetUploadState(LONG IUploadHandle, LPDWORD pProgress)

参 数： [in] IUploadHandle 文件上传的句柄，NET_DVR_UploadFile_V40 的返回值
[out] pProgress 返回的进度值，取值范围：0~100

返回值： -1 表示函数调用失败，其他为上传的状态值：1- 上传成功；2- 正在上传；3- 上传失败；4- 网络断开，状态未知。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](#)

6.28.23 停止文件上传 **NET_DVR_UploadClose**

函 数： BOOL NET_DVR_UploadClose(LONG IUploadHandle)

参 数： [in] IUploadHandle 文件上传的句柄，NET_DVR_UploadFile_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](#)

6.28.24 开始下载文件 **NET_DVR_StartDownload**

函 数： LONG NET_DVR_StartDownload(LONG IUserID, DWORD dwDownloadType, LPVOID lpInBuffer, DWORD dwInBufferSize, char const *sFileName)

参 数： [in] IUserID 登录主控板，NET_DVR_Login_V40 的返回值
[in] dwDownloadType 下载文件类型，详见表 6.120
[in] lpInBuffer 输入参数。不同的 dwUploadType，输入参数不同，详见表 6.120
[in] dwInBufferSize 输入缓冲区大小
[in] sFileName 下载文件的保存路径（绝对路径，包括文件名）

返回值： -1 表示失败，其他值作为 NET_DVR_StopDownload 和 NET_DVR_GetDownloadState 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：

表 6.120 文件下载类型

dwDownloadType 宏定义	宏定义值	dwDownloadType 含义	lpInBuffer 对应结构体
NET_SDK_DOWNLOAD_CONFIGURATION_FILE	4	下载配置文件	NULL

[返回目录](#)

6.28.25 获取文件下载的进度和状态 **NET_DVR_GetDownloadState**

函 数： LONG NET_DVR_GetDownloadState(LONG IDownloadHandle, LPDWORD pProgress)

参 数： [in] IDownloadHandle 文件下载的句柄，NET_DVR_StartDownload 的返回值
[out] pProgress 返回的进度值，取值范围：0~100

返回值： -1 表示函数调用失败，其他为下载的状态值：1- 下载成功；2- 正在下载；3- 下载失败；4- 网络断开，状态未知。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](#)

6.28.26 停止文件下载 **NET_DVR_StopDownload**

函 数： BOOL NET_DVR_StopDownload(LONG IHandle);

参 数： [in] IHandle 文件下载的句柄，NET_DVR_StartDownload 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](#)

7 错误代码及说明

7.1 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_DEVICETYPE_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_ERR_VIRTUAL_LED_VERTICAL_CROSS	729	虚拟 LED 不能纵向跨屏
NET_ERR_MAX_VIRTUAL_LED_HEIGHT	730	虚拟 LED 高度超限
NET_ERR_VIRTUAL_LED_ILLEGAL_CHARACTER	731	虚拟 LED 内容包含非法字符
NET_ERR_BASEMAP_NOT_EXIST	732	底图图片不存在
NET_ERR_LED_NOT_SUPPORT_VIRTUAL_LED	733	LED 屏幕不支持虚拟 LED
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_CUT_INPUTSTREAM_OVERLIMIT	900	信号源裁剪数值超
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
NET_ERR_SIGNAL_MAX_ENLARGE_TIMES	928	信号源放大倍数超限
NET_ERR_ONE_SIGNAL_MULTI_CROSS	929	单个信号源不能多次跨屏
NET_ERR_ULTRA_HD_SIGNAL_MULTI_WIN	930	超高清信号源不能重复开窗
NET_ERR_MAX_VIRTUAL_LED_WIDTH	931	虚拟 LED 宽度大于限制值
NET_ERR_MAX_VIRTUAL_LED_WORD_LEN	932	虚拟 LED 字符数大于限制值
NET_ERR_SINGLE_OUTPUTPARAM_CONFIG	933	不支持单个显示输出参数设置
NET_ERR_MULTI_WIN_BE_COVER	934	多分屏窗口被覆盖
NET_ERR_WIN_NOT_EXIST	935	窗口不存在
NET_ERR_WIN_MAX_SIGNALSOURCE	936	窗口信号源数超过限制值
NET_ERR_MULTI_WIN_MOVE	937	对多分屏窗口移动
NET_ERR_MULTI_WIN_YBPBR_SDI	938	YPBPR 和 SDI 信号源不支持 9/16 分屏
NET_ERR_DIFF_TYPE_OUTPUT_MIXUSE	939	不同类型输出板混插
NET_ERR_SPLIT_WIN_CROSS	940	对跨屏窗口分屏
NET_ERR_SPLIT_WIN_NOT_FULL_SCREEN	941	对未全屏窗口分屏
NET_ERR_SPLIT_WIN_MANY_WIN	942	对单个输出口上有多个窗口的窗口分屏
NET_ERR_WINDOW_SIZE_OVERLIMIT	943	窗口大小超限
NET_ERR_INPUTSTREAM_ALREADY_JOINT	944	信号源已加入拼接
NET_ERR_JOINT_INPUTSTREAM_OVERLIMIT	945	拼接信号源个数超限
NET_ERR_LED_RESOLUTION	946	LED 分辨率大于输出分辨率
NET_ERR_JOINT_SCALE_OVERLIMIT	947	拼接信号源的规模超限
NET_ERR_INPUTSTREAM_ALREADY_DECODE	948	信号源已上墙
NET_ERR_INPUTSTREAM_NOTSUPPORT_CAPTURE	949	信号源不支持抓图
NET_ERR_JOINT_NOTSUPPORT_SPLITWIN	950	拼接信号源不支持分屏
NET_ERR_CLOSE_WINDOW_FIRST	971	切换全帧率畅显时必须先关闭对应的已上墙的窗口
NET_ERR_SPLIT_WINDOW_NUM_NOT_SUPPORT	972	DVI/DP/HDMI/HDBase_T 输入源在全帧率畅显下不支持 9/16 画面
NET_ERR_REACH_ONE_SIGNAL_PREVIEW_MAX_LINK	973	单信号源回显连接数量超限
NET_ERR_ONLY_SPLITWND_SUPPORT_AMPLIFICATION	974	只有分屏窗口支持子窗口放大
NET_DVR_ERR_WINDOW_SIZE_PLACE	975	窗口位置错误
NET_DVR_ERR_REGIONAL_RESTRICTIONS	976	屏幕距离超限
NET_ERR_WNDZOOM_NOT_SUPPORT	977	单窗口不支持子窗口全屏功能
NET_ERR_LED_SCREEN_SIZE	978	LED 屏宽高不正确
NET_ERR_OPEN_WIN_IN_ERROR_AREA	979	在非法区域开窗(包括跨 LCD/LED 屏)
NET_ERR_TITLE_WIN_NOT_SUPPORT_MOVE	980	平铺模式不支持漫游
NET_ERR_TITLE_WIN_NOT_SUPPORT_COVER	981	平铺模式不支持图层覆盖
NET_ERR_TITLE_WIN_NOT_SUPPORT_SPLIT	982	平铺模式不支持分屏
NET_DVR_LED_WINDOWS_ALREADY_CLOSED	983	LED 区域内输出口的分辨率发生变化，设备已关闭该区域内的所有 LED 窗口
NET_DVR_ERR_CLOSE_WINDOWS	984	操作失败，请先关闭窗口
NET_DVR_ERR_MATRIX_LOOP_ABILITY	985	超出轮巡解码能力限制

NET_DVR_ERR_MATRIX_LOOP_TIME	986	轮巡解码时间不支持
NET_DVR_ERR_LINKED_OUT_ABILITY	987	联动通道数超过上限
NET_ERR_REACH_SCENE_MAX_NUM	988	场景数量达到上限
NET_ERR_SCENE_MEM_NOT_ENOUGH	989	内存不足，无法新建场景
NET_ERR_RESOLUTION_NOT_SUPPORT_ODD_VOUT	990	奇口不支持该分辨率
NET_ERR_RESOLUTION_NOT_SUPPORT_EVEN_VOUT	991	偶口不支持该分辨率

7.2 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_SETUPRECVDATALOST	424	RTSP SETUP 接收数据错误
NET_DVR_RTSP_SETUPRECVDERROR	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_PLAYRECVDATALOST	434	RTSP PLAY 接收数据错误
NET_DVR_RTSP_PLAYRECVDERROR	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_TEARDOWNRECVDATALOST	444	RTSP TEARDOWN 接收数据错误
NET_DVR_RTSP_TEARDOWNRECVDERROR	445	RTSP TEARDOWN 接收失败
NET_DVR_RTSP_TEARDOWNSERVERERR	446	RTSP TEARDOWN 设备返回错误状态

7.3 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

8 结构体说明

8.1 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_SUBSYSTEM_NUM_V40	120	视频综合平台中最大子系统个数（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	设备可配以太网网络个数
MAX_DECODECHANNUM	32	多路解码器最大解码通道数
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_TIMESEGMENT_V30	8	最大时间段数
MAX_EXCEPTIONNUM	16	设备最大异常处理数
MAX_EXCEPTIONNUM_V30	32	设备最大异常处理数（扩展）
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_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）
DESC_LEN	16	云台描述字符串长度
MAX_DISPLAY_NUM	512	最大显示输出个数
MAX_SUBBOARD_NUM	42	集中式大屏设备子板最大数目
MAX_SINGLE_BOARD_EXCEPTION_NUM	16	单板最大并发异常数

MAX_SERVERID_LEN	64	最大服务器 ID 的长度
MAX_AUTHENTICATEID_LEN	64	认证 ID 最大长度
MAX_AUTHENTICATEPASSWD_LEN	32	认证密码最大长度
MAX_SERVERNAME_LEN	64	最大服务器用户名
MAX_COMPRESSIONID_LEN	64	编码 ID 的最大长度
MAX_SIPSERVER_ADDRESS_LEN	128	SIP 服务器地址支持域名和 IP 地址
MAX_OPTICAL_DEV_NODE	32	最多节点光端机数
MAX_LEN_OSD_CONTENT	256	OSD 信息最大长度
MAX_NUM_OSD_ONE_SUBWND	8	单个子窗口支持的最大 OSD 数量
MAX_NUM_SPLIT_WND	64	单个窗口支持的最大分屏窗口数量（即子窗口数量）
MAX_NUM_OUTPUT_CHANNEL	512	LED 区域包含的最大输出口个数
MAX_VS_INPUT_CHAN_NUM	16	虚拟屏输入通道最大数量
MAX_RES_NUM_ONE_VS_INPUT_CHAN	8	一个虚拟屏输入通道支持的分辨率的最大数量
FILE_NAME_LEN	32	文件名称最大长度

8.2 设备型号

宏定义	宏定义值	设备类型
DS_B10_XY	100	视频综合平台 B10 设备型号
DS_6504HF_B10	101	视频综合平台 B10 内部编码器
DS_6504D_B10	102	视频综合平台 B10 内部解码器
DS_1832_B10	103	视频综合平台 B10 内部码分器
DS_6401HFH_B10	104	视频综合平台 B10 内部光纤板
DS_1964_B10	121	视频综合平台 B10 内部报警器
DS_B10N04_IN	122	视频综合平台 B10 内部级联输入
DS_B10N04_OUT	123	视频综合平台 B10 内部级联输出
DS_B10N04_INTEL	124	视频综合平台 B10 内部智能
DS_6408HFH_B10E_RM	125	B10 V6 高清
DS_B10N64F1_RTM	126	B10 V6 级联不带 DSP
DS_B10N64F1D_RTM	127	B10 V6 级联带 DSP
DS_B10_SDS	128	B10 子域
DS_B10_DS	129	B10 域
DS_6401HFH_B10V	130	B10 VGA 高清编码器
DS_6504D_B10B	131	视频综合平台 B10 内部标清解码器

DS_6504D_B10H	132	视频综合平台 B10 内部高清解码器
DS_6504D_B10V	133	视频综合平台 B10 内部 VGA 解码器
DS_6408HFH_B10S	134	视频综合平台 B10 SDI 子板
DS_18XX_N	135	矩阵接入网关
DS_6504HF_B10F_CLASS	136	标清光端机接入编码器（支持 4/8/12 路光端机接入）
DS_B11_M_CLASS	301	视频综合平台 B11 设备型号
DS_B12_M_CLASS	302	视频综合平台 B12 设备型号
DS_6504HF_B11_CLASS	303	视频综合平台 B11 内部编码器
DS_6504HF_B12_CLASS	304	视频综合平台 B12 内部编码器
DS_6401HFH_B11V_CLASS	305	B11 VGA 高清
DS_6401HFH_B12V_CLASS	306	B12 VGA 高清
DS_6408HFH_B11S_CLASS	307	B11 SDI
DS_6408HFH_B12S_CLASS	308	B12 SDI
DS_6504D_B11H_CLASS	309	视频综合平台 B11 内部高清解码器
DS_6504D_B11B_CLASS	310	视频综合平台 B11 内部标清解码器
DS_6504D_B12B_CLASS	311	视频综合平台 B12 内部标清解码器
DS_6504D_B11V_CLASS	312	视频综合平台 B11 内部 VGA 解码器*
DS_6504D_B12V_CLASS	313	视频综合平台 B12 内部 VGA 解码器
DS_6401HFH_B10R_CLASS	314	B10 RGB 高清
DS_6401HFH_B10D_CLASS	315	B10 DVI 高清
DS_6401HFH_B10H_CLASS	316	B10 HDMI 高清
DS_6401HFH_B11R_CLASS	317	B11 RGB 高清
DS_6401HFH_B11D_CLASS	318	B11 DVI 高清
DS_6401HFH_B11H_CLASS	319	B11 HDMI 高清
DS_6401HFH_B12R_CLASS	320	B12 RGB 高清
DS_6401HFH_B12D_CLASS	321	B12 DVI 高清
DS_6401HFH_B12H_CLASS	322	B12 HDMI 高清
DS_65XXD_B10Ex_CLASS	323	netra 高清解码器
DS_6516HW_B10_CLASS	324	模拟高清（960H）视频输入板（BNC 接口）
DS_6408HFH_B10F_RX_CLASS	326	高清光端机接入编码器（支持 1/2 路光端机接入）
DS_6524HW_B10F_RX_CLASS	327	960H 光端机接入编码器（支持 1/4/8 路光端机接入）
DS_6504D_B11Ex_CLASS	328	B11 Netra 高清解码器
DS_6504D_B12Ex_CLASS	329	B12 Netra 高清解码器
DS_6512HW_B11_CLASS	330	B11 Netra 高线编码器
DS_6512HW_B12_CLASS	331	B12 Netra 高线编码器

DS_6504D_B10H_CLASS	332	视频综合平台 B10 内部高清解码器
DS_65XXT_B10_CLASS	333	视频综合平台 B10 转码子系统
DS_6508HFH_B10ES_CLASS	338	B10 SDI 输入编码子系统
DS_6402HFH_B10EV_CLASS	345	B10 Netra VGA 编码板
DS_6402HFH_B10ED_CLASS	346	B10 Netra DVI 编码板
DS_6402HFH_B10EH_CLASS	347	B10 Netra HDMI 编码板
DS_6404HFH_B10T_RX_CLASS	348	B10 光纤接入编码板
DS_6508HF_B10E_CLASS	355	B10 BNC 输入编码子系统
DS_6404HFH_B10ES_CLASS	356	B10 SDI 输入编码子系统
DS_6402HFH_B10ER_CLASS	358	B10 RGB 输入编码子系统
DS_6404HFH_B10T_RM_CLASS	361	B10 光纤输入编码子系统
DS_6516D_B10EB_CLASS	362	B10 BNC 输出解码子系统
DS_6516D_B10ES_CLASS	363	B10 SDI 输出解码子系统
DS_6508HF_B10E_CLASS	355	B10 BNC 输入编码子系统
DS_6404HFH_B10ES_CLASS	356	B10 SDI 输入编码子系统
DS_6402HFH_B10ER_CLASS	358	B10 RGB 输入编码子系统
DS_6404HFH_B10T_RM_CLASS	361	B10 光纤输入编码子系统
DS_6516D_B10EB_CLASS	362	B10 BNC 输出解码子系统
DS_6516D_B10ES_CLASS	363	B10 SDI 输出解码子系统
DS_6508_B11E_CLASS	365	B11 BNC 输入编码子系统
DS_6402_B11ES_CLASS	366	B11 SDI 输入编码子系统
DS_6402_B11EV_CLASS	367	B11 VGA 输入编码子系统
DS_6402_B11ER_CLASS	368	B11 RGB 输入编码子系统
DS_6402_B11ED_CLASS	369	B11 DVI 输入编码子系统
DS_6402_B11EH_CLASS	370	B11 HDMI 输入编码子系统
DS_6516D_B11EB_CLASS	371	B11 BNC 输出解码子系统
DS_6516D_B11ES_CLASS	372	B11 SDI 输出解码子系统
DS_6508_B12E_CLASS	373	B12 BNC 输入编码子系统
DS_6402_B12ES_CLASS	375	B12 SDI 输入编码子系统
DS_6402_B12EV_CLASS	376	B12 VGA 输入编码子系统
DS_6402_B12ER_CLASS	377	B12 RGB 输入编码子系统
DS_6402_B12ED_CLASS	378	B12 DVI 输入编码子系统
DS_6402_B12EH_CLASS	379	B12 HDMI 输入编码子系统
DS_6516D_B12EB_CLASS	380	B12 BNC 输出解码子系统
DS_B20_MSU_NP	400	B20 主控板

DS_6416HFH_B20S	401	B20SDI 输入编码
DS_6416HFH_B20_RM	402	B20 光纤输入高清编码
DS_6564D_B20D	403	B20 DVI 解码
DS_6564D_B20H	404	B20 HDMI 解码
DS_6564D_B20V	405	B20 VGA 解码
DS_B20_6516D_DEV_CLASS	406	B20 解码子系统
DS_6408HFH_B20V	407	B20 VGA 编码板
DS_MMC_B20_CLASS	408	B20 主控
DS_CARD_CHIP_B20_CLASS	409	B20 主控子板
DS_6564D_B20B_DEV_CLASS	410	B20 BNC 解码子系统
DS_6564D_B20S_DEV_CLASS	411	B20 SDI 解码子系统
DS_6532HF_B20B_DEV_CLASS	412	B20 BNC 编码子系统
DS_6408HFH_B20D_DEV_CLASS	413	B20 DVI 编码子系统
DS_6408HFH_B20H_DEV_CLASS	414	B20 HDMI 编码子系统
DS_IVMSE_B20_CLASS	415	B20 X86 服务器子系统
DS_6402HFH_B20Y_DEV_CLASS	416	B20 YUV 编码子系统
DS_6508HW_B20_DEV_CLASS	417	B20 HW 编码子系统
DS_B20N128Fx_B20_DEV_CLASS	418	B20 DS_B20N128Fx_M 级联板
DS_AIO_MCU_NP_DEV_CLASS	419	视频综合平台一体机 IO 主控板
DS_6402_AIO_EV_DEV_CLASS	420	视频综合平台一体机 VGA 编码
DS_6508D_AIO_EV_DEV_CLASS	421	视频综合平台一体机 VGA 解码
DS_6508D_AIO_ED_DEV_CLASS	422	视频综合平台一体机 DVI 解码
DS_6508D_AIO_EH_DEV_CLASS	423	视频综合平台一体机 HDMI 解码
DS_6508HD_B20F_DEV_CLASS	424	B20 视频增强板
DS_6402HFH_B20ES_DEV_CLASS	425	B20 3G SID 编码
DS_6532D_B20_DEV_CLASS	426	B20 解码子系统
DS_IVMST_B20_DEV_CLASS	427	B20 X86 服务器子系统
DS_6416HFH_B20DD_DEV_CLASS	428	B20 VI 双链路
DS_6441VS_B20_DEV_CLASS	429	B20 相机拼接类型
DS_B21_MCU_NP_CLASS	440	B21 主控
DS_B21_S10_x_CLASS	441	B21 机箱, x = A/S/D
DS_6402HFH_B21D_CLASS	442	B21 编码子系统
DS_6508HD_B21D_CLASS	443	B21 解码子系统
DS_iVMSE_B21HW_CLASS	444	B21 X86 子系统
DS_C10S	501	C10S 集中式大屏控制器

8.3 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_V30 或 NET_DVR_Login_V40 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

8.4 NET_DVR_ALLSUBSYSTEMINFO_V40:视频综合平台所有子系统参数

```
struct{
    DWORD                dwSize;
    NET\_DVR\_SUBSYSTEMINFO\_V40 struSubSystemInfo[MAX_SUBSYSTEM_NUM_V40];
    BYTE                 byRes[8];
}NET_DVR_ALLSUBSYSTEMINFO_V40, *LPNET_DVR_ALLSUBSYSTEMINFO_V40;
```

Members***dwSize***

结构体大小

struSubSystemInfo

子系统参数结构体

byRes

保留

8.5 NET_DVR_ASSOCIATE_INPUT_PARAM:关联的输入源参数

```
struct{
    BYTE    byEnable;
    BYTE    byRes1[3];
    DWORD   dwDeviceIndex;
    WORD    wInputIndex;
    BYTE    byRes2[18];
}NET_DVR_ASSOCIATE_INPUT_PARAM, *LPNET_DVR_ASSOCIATE_INPUT_PARAM;
```

Members

byEnable

使能：0- 禁用，1- 启用

byRes1

保留

dwDeviceIndex

分布式大屏控制器设备序号

wInputIndex

输入信号源

byRes2

保留

Remarks

当设备为集中式多屏控制器使用，用信号源序号（wInputIndex）表示输入设备。

8.6 NET_DVR_ASSOCIATE_OUTPUT_PARAM: 关联的输出显示屏信息

```

struct{
    BYTE      byEnable;
    BYTE      byWallNo;
    BYTE      byRes1[2];
    DWORD     dwOutputIndex;
    BYTE      byRes2[20];
}NET_DVR_ASSOCIATE_OUTPUT_PARAM,*LPNET_DVR_ASSOCIATE_OUTPUT_PARAM;

```

Members*byEnable*

是否启用：0- 否，1- 是

byWallNo

电视墙序号

byRes1

保留

dwOutputIndex

输出显示屏的序号，从 1 开始

byRes2

保留

8.7 NET_DVR_ASSOCIATEDDEV_CHAN_INFO: 窗口关联的编码设备信息

```

struct{
    DWORD     dwSize;
    BYTE      sAddress[MAX_DOMAIN_NAME];
    WORD      wDVRPort;
    WORD      wChannel;
    BYTE      sUserName[NAME_LEN];
    BYTE      sPassword[PASSWD_LEN];
}

```

```

    BYTE        byRes[24];
}NET_DVR_ASSOCIATEDDEV_CHAN_INFO,*LPNET_DVR_ASSOCIATEDDEV_CHAN_INFO;

```

Members

dwSize

结构体大小

sAddress

设备 IP 地址或域名

wDVRPort

端口号

wChannel

通道号

sUserName

监控主机登录帐号

sPassword

监控主机密码

byRes

保留

8.8 NET_DVR_AUDIO_CHAN_CFG:音频切换配置

```

struct{
    DWORD                dwSize;
    BYTE                 sChanName[NAME_LEN];
    BYTE                 byEnable;
    BYTE                 byAudioSwitchType;
    BYTE                 byRes[2];
    NET\_DVR\_PU\_STREAM\_CFG\_V41 struAudioSrcInfo;
    DWORD                dwWindowNo;
    BYTE                 byRes2[28];
}NET_DVR_AUDIO_CHAN_CFG,*LPNET_DVR_AUDIO_CHAN_CFG;

```

Members

dwSize

结构体大小

sChanName

音频通道名称

byEnable

音频是否开启：0- 关，1- 开

byAudioSwitchType

音频切换方式：1- 按前端源方式，2- 按窗口解码源方式

byRes

保留，置为 0

struAudioSrcInfo

音频源信息，byAudioSwitchType 为 1 时有效

dwWindowNo

子窗口号(组合): 1 字节墙号+1 字节子窗口号+2 字节窗口号, byAudioSwitchType 为 2 时有效
byRes2
 保留, 置为 0

8.9 NET_DVR_AUDIO_CHAN_INFO:音频通道信息

```
struct{
    DWORD    dwSize;
    DWORD    dwChannel;
    BYTE     byRes[48];
}NET_DVR_AUDIO_CHAN_INFO, *LPNET_DVR_AUDIO_CHAN_INFO;
```

Members

dwSize

结构体大小

dwChannel

音频通道号(组合): 1 字节设备号+1 字节子板号+2 字节音频通道号

byRes

保留, 置为 0

Remarks

- 对于视频综合平台, 设备号为 1, 子板号和音频通道号通过 [NET_DVR_MatrixGetSubSystemInfo V40](#) 获取 (对应 [NET_DVR_SUBSYSTEMINFO V40](#) 里面的 byBelongBoard、byAudioChanNums、byAudioChanStartNum、byAudioChanType), 需要将音频界面输出哪块板的哪个输出口, 就选择对应的编号。
- 对应 69UD 解码器, 设备号为 1, 子板号为 0。而音频输出口跟 HDMI 类型的视频输出口一一对应, 所以音频通道号跟显示输出口号一致, 显示输出口号可以通过 [NET_DVR_GetDVRConfig](#)(命令: NET_DVR_GET_VIDEOWALLDISPLAYNO)获取, 取其中的 HDMI 类型的输出口号。

8.10 NET_DVR_BASEMAP_CFG:底图参数

```
struct{
    BYTE     byScreenIndex;
    BYTE     byMapNum;
    BYTE     res[2];
    WORD     wSourWidth;
    WORD     wSourHeight;
}NET_DVR_BASEMAP_CFG, *LPNET_DVR_BASEMAP_CFG;
```

Members

byScreenIndex

屏幕的序号

byMapNum

被分割成了多少块, 1 表示原始大小。第一次传递数据, 该值必须是 0

res

保留

wSourWidth

切割后图片的宽度，byMapNum 为 1 时表示原始宽度

wSourHeight

切割后图片的高度，byMapNum 为 1 时表示原始高度

8.11 NET_DVR_BASEMAP_CONTROL_CFG:底图参数配置

```
struct{
    DWORD    dwSize;
    BYTE      byEnable;
    BYTE      byBaseMapType;
    BYTE      byRes1[2];
    DWORD    dwBaseMapNo;
    BYTE      byRes2[64];
}NET_DVR_BASEMAP_CONTROL_CFG,*LPNET_DVR_BASEMAP_CONTROL_CFG;
```

Members

dwSize

结构体大小

byEnable

是否显示：1-显示，0-隐藏

byBaseMapType

底图类型：1-图片底图，2-超高清输入底图

byRes1

保留

dwBaseMapNo

底图号：底图类型为 1 时，此参数为图片序号；类型为 2 时，此参数为超高清输入子系统输入口号 (组合)：1 字节矩阵号+1 字节子板号+2 字节输入口号

byRes2

保留

8.12 NET_DVR_BASEMAP_PIC_INFO:底图图片信息

```
struct{
    DWORD    dwSize;
    BYTE      sPicName[NAME_LEN];
    BYTE      byUsed;
    BYTE      byRes[31];
}NET_DVR_BASEMAP_PIC_INFO,*LPNET_DVR_BASEMAP_PIC_INFO;
```

Members

dwSize

结构体大小

sPicName

图片名称

byUsed

是否已存在：0-不存在，1-存在

byRes

保留

8.13 NET_DVR_BASEMAP_WINCFG:底图窗口参数

```
struct{
    DWORD                dwSize;
    BYTE                 byEnable;
    BYTE                 byRes1[7];
    NET\_DVR\_RECTCFG\_EX  struWinPosition;
    BYTE                 byRes2[64];
}NET_DVR_BASEMAP_WINCFG, *LPNET_DVR_BASEMAP_WINCFG;
```

Members

dwSize

结构体大小

byEnable

是否启用：0- 否，1- 是

byRes1

保留

struWinPosition

目的窗口(相对显示墙)，底图窗口的起始坐标和大小必须是基准坐标的整数倍

byRes2

保留

Remarks

底图窗口不能重叠，可用此接口调整底图窗口位置，不支持动态改变窗口大小。

8.14 NET_DVR_BOARDALARMCFG:热插拔报警配置参数

```
struct{
    BYTE                 byEnablePullAlarm;
    BYTE                 byRes1[3];
    NET\_DVR\_HANDLEEXCEPTION\_V30 struTempHandleType;
    NET\_DVR\_SCHEDTIME      struAlarmTime[MAX_DAYS][MAX_TIMESEGMENT_V30];
    BYTE                 byRes2[32];
}NET_DVR_BOARDALARMCFG, *LPNET_DVR_BOARDALARMCFG;
```

Members

byEnablePullAlarm

热插拔报警使能：0-报警，1-不报警

byRes1

保留

struTempHandleType

处理方式

struAlarmTime
布防时间

byRes2
保留

8.15 NET_DVR_BUF_INFO:缓冲区信息

```
struct{
    void      *pBuf;
    DWORD     nLen;
}NET_DVR_BUF_INFO, *LPNET_DVR_BUF_INFO;
```

Members

pBuf
缓冲区指针

nLen
缓冲区长度

8.16 NET_DVR_CAMERACHAN_SERIALCFG: Camera 通道绑定串口配置

```
struct{
    DWORD     dwSize;
    DWORD     dwBaudRate;
    BYTE      byDataBit;
    BYTE      byStopBit;
    BYTE      byParity;
    BYTE      byFlowcontrol;
    WORD      wPTZProtocol;
    BYTE      byRes1[6];
    DWORD     dwSerialPort;
    BYTE      bySerialAddress;
    BYTE      byRes[15];
}NET_DVR_CAMERACHAN_SERIALCFG, *LPNET_DVR_CAMERACHAN_SERIALCFG;
```

Members

dwSize
结构体大小

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

wPTZProtocol

控球协议类型，索引值，设置支持的协议通过 [NET_DVR_GetPTZProtocol_Ex](#) 获取

byRes1

保留，置为 0

dwSerialPort

串口号，最高位为 0，表示前面板串口，最高位为 1，表示后面板串口

bySerialAddress

串口地址 0~255

byRes

保留，置为 0

8.17 NET_DVR_CHAN_INFO:设备通道信息

```
struct{
    DWORD          dwSize;
    BYTE           byValid;
    BYTE           byRes1[3];
    BYTE           sChanName[NAME_LEN];
    NET\_DVR\_COLOR  struVideoColor;
    WORD           wResolutionX;
    WORD           wResolutionY;
    BYTE           byRes2[40];
}NET_DVR_CHAN_INFO,*LPNET_DVR_CHAN_INFO;
```

Members

dwSize

结构体大小

byValid

是否有效：0- 无效，1- 有效

byRes1

保留

sChanName

通道名称

struVideoColor

信号源颜色

wResolutionX

当前分辨率的宽度

wResolutionY

当前分辨率的高度

byRes2

保留

8.18 NET_DVR_CODEDEV_INFO:编码子设备信息

```
struct{
    NET_DVR_CHAN_INFO    struChanInfo[MAX_CHANNUM];
}NET_DVR_CODEDEV_INFO,*LPNET_DVR_CODEDEV_INFO;
```

Members

struChanInfo
设备通道信息

8.19 NET_DVR_COLOR:图像参数

```
struct{
    BYTE    byBrightness;
    BYTE    byContrast;
    BYTE    bySaturation;
    BYTE    byHue;
}NET_DVR_COLOR, *LPNET_DVR_COLOR;
```

Members

byBrightness
亮度，取值范围[0,255]

byContrast
对比度，取值范围[0,255]

bySaturation
饱和度，取值范围[0,255]

byHue
色调，取值范围[0,255]

8.20 NET_DVR_CONTROL_PARAM:控制参数

```
struct{
    DWORD    dwSize;
    BYTE     sDeviceID[NAME_LEN];
    WORD     wChan;
    BYTE     byIndex;
    BYTE     byRes1;
    DWORD    dwControlParam;
    BYTE     byRes2[32];
}NET_DVR_CONTROL_PARAM,*LPNET_DVR_CONTROL_PARAM;
```

Members

dwSize
结构体大小

sDeviceID

被控设备的设备 ID，设为 NULL

wChan

控制预案时有效，表示电视墙号，从 1 开始

byIndex

控制索引，不同的控制命令对应不同的索引：

dwCommand 控制命令	控制功能描述	byIndex 索引
NET_DVR_OPEN_PLAN	控制预案	预案号，从 1 开始，取值范围可以通过电视墙能力集获取

byRes1

保留，置为 0

dwControlParam

控制参数，不同的控制命令对应不同的参数：

dwCommand 控制命令	控制功能描述	dwControlParam 参数
NET_DVR_OPEN_PLAN	控制预案	1- 预案开始，2- 预案停止

byRes2

保留，置为 0

8.21 NET_DVR_CYCLE_TIME: 预案循环时间参数

```
struct{
    BYTE                byValid;
    BYTE                byRes[3];
    NET\_DVR\_TIME\_EX    struTime;
}NET_DVR_CYCLE_TIME,*LPNET_DVR_CYCLE_TIME;
```

Members

byValid

是否有效

byRes

保留，置为 0

struTime

时间

8.22 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

8.23 NET_DVR_DEC_DDNS_DEV:动态域名取流配置

```

struct{
    NET\_DVR\_DEV\_DDNS\_INFO          struDdnsInfo;
    NET\_DVR\_STREAM\_MEDIA\_SERVER    struMediaServer;
}NET_DVR_DEC_DDNS_DEV,*LPNET_DVR_DEC_DDNS_DEV;

```

Members*struDdnsInfo*

动态域名参数配置

struMediaServer

流媒体服务器配置参数

8.24 NET_DVR_DEC_STREAM_DEV:设备取流配置

```

struct{
    NET\_DVR\_STREAM\_MEDIA\_SERVER\_CFG\_V41    struStreamMediaSvrCfg;
    NET\_DVR\_DEV\_CHAN\_INFO\_V41              struDevChanInfo;
}NET_DVR_DEC_STREAM_DEV,*LPNET_DVR_DEC_STREAM_DEV;

```

Members*struStreamMediaSvrCfg*

流媒体服务器配置参数

struDevChanInfo

设备通道配置参数

8.25 NET_DVR_DEC_STREAM_DEV_EX:设备取流配置

```
struct{
    NET\_DVR\_STREAM\_MEDIA\_SERVER    struStreamMediaSvrCfg;
    NET\_DVR\_DEV\_CHAN\_INFO\_EX        struDevChanInfo;
}NET_DVR_DEC_STREAM_DEV_EX,*LPNET_DVR_DEC_STREAM_DEV_EX;
```

Members

struStreamMediaSvrCfg

流媒体服务器配置参数

struDevChanInfo

设备通道配置参数

8.26 NET_DVR_DEC_STREAM_MODE:取流模式配置

```
union{
    NET\_DVR\_DEC\_STREAM\_DEV\_EX    struDecStreamDev;
    NET\_DVR\_PU\_STREAM\_URL        struUrlInfo;
    NET\_DVR\_DEC\_DDNS\_DEV         struDdnsDecInfo;
    BYTE                         byRes[300];
}NET_DVR_DEC_STREAM_MODE,*LPNET_DVR_DEC_STREAM_MODE;
```

Members

struDecStreamDev

通过 IP 地址或者域名从设备或者流媒体服务器取流

struUrlInfo

通过 URL 从设备或者流媒体服务器取流

struDdnsDecInfo

通过动态域名解析向设备取流

byRes

保留，置为 0

8.27 NET_DVR_DECODEDEV_INFO:解码子设备信息

```
struct{
    BYTE    byRes[1408];
}NET_DVR_DECODEDEV_INFO,*LPNET_DVR_DECODEDEV_INFO;
```

Members

byRes

保留

8.28 NET_DVR_DEV_BASE_INFO:注册设备信息

```
struct{
    DWORD    dwSize;
    BYTE      byEnable;
    BYTE      byDeviceType;
    BYTE      byRes1[2];
    BYTE      sDevName[NAME_LEN];
    BYTE      byRes2[24];
}NET_DVR_DEV_BASE_INFO,*LPNET_DVR_DEV_BASE_INFO;
```

Members

dwSize

结构体大小

byEnable

是否在系统中使用：0- 否，1- 是

byDeviceType

设备类型(只能获取)，具体定义如下：

```
enum{
    NET_DVR_BNC_PROC    = 1,
    NET_DVR_RGB_PROC    = 2,
    NET_DVR_STP_PROC    = 3,
    NET_DVR_DISP_PROC   = 4,
    NET_DVR_NETSIGNAL   = 5,
    NET_DVR_SDI_PROC    = 6
}NET_DVCS_DEVICE_TYPE
```

NET_DVR_BNC_PROC

BNC 处理器

NET_DVR_RGB_PROC

RGB 处理器

NET_DVR_STP_PROC

转码处理器

NET_DVR_DISP_PROC

显示处理器

NET_DVR_NETSIGNAL

网络信号源

NET_DVR_SDI_PROC

SDI 处理器

byRes1

保留

sDevName

设备名称

byRes2

保留

8.29 NET_DVR_DEV_CHAN_INFO:前端编码设备信息

```
struct{
    NET_DVR_IPADDR    struIP;
    WORD              wDVRPort;
    BYTE              byChannel;
    BYTE              byTransProtocol;
    BYTE              byTransMode;
    BYTE              byFactoryType;
    BYTE              byDeviceType;
    BYTE              byDispChan;
    BYTE              bySubDispChan;
    BYTE              byResolution;
    BYTE              byRes[2];
    BYTE              byDomain[MAX_DOMAIN_NAME];
    BYTE              sUserName[NAME_LEN];
    BYTE              sPassword[PASSWD_LEN];
}NET_DVR_DEV_CHAN_INFO,*LPNET_DVR_DEV_CHAN_INFO;
```

Members

struIP

设备 IP 地址

wDVRPort

设备端口号

byChannel

通道号,目前设备的模拟通道号是从 1 开始的,对于 9000 等设备的 IPC 接入,数字通道号从 33 开始

byTransProtocol

传输协议类型: 0-TCP, 1-UDP, 2-多播方式, 3-RTP

byTransMode

传输码流模式: 0- 主码流, 1- 子码流

byFactoryType

前端设备厂家类型, 通过接口 [NET_DVR_GetIPCProtoList_V41](#) 获取

byDeviceType

设备类型(视频综合平台使用): 1- IPC, 2- ENCODER

byDispChan

显示通道号 (智能配置使用), 根据能力集决定使用解码通道还是显示通道

bySubDispChan

显示通道子通道号 (智能配置时使用)

byResolution

分辨率: 1- CIF, 2- 4CIF, 3- 720P, 4- 1080P, 5- 500W, 用于多屏控制器, 多屏控制器会根据该参数分配解码资源

byRes

保留, 置为 0

byDomain

设备域名

sUserName

设备登陆帐号

sPassword

设备密码

8.30 NET_DVR_DEV_CHAN_INFO_EX:前端编码设备信息

```
struct{
    BYTE        byChanType;
    BYTE        byStreamId[STREAM_ID_LEN];
    BYTE        byRes1[3];
    DWORD       dwChannel;
    BYTE        byRes2[24];
    BYTE        byAddress[MAX_DOMAIN_NAME];
    WORD        wDVRPort;
    BYTE        byChannel;
    BYTE        byTransProtocol;
    BYTE        byTransMode;
    BYTE        byFactoryType;
    BYTE        byDeviceType;
    BYTE        byDispChan;
    BYTE        bySubDispChan;
    BYTE        byResolution;
    BYTE        byRes[2];
    BYTE        sUserName[NAME_LEN];
    BYTE        sPassword[PASSWD_LEN];
}NET_DVR_DEV_CHAN_INFO_EX,*LPNET_DVR_DEV_CHAN_INFO_EX;
```

Members

byChanType

通道类型：0- 普通通道，1- 零通道，2- 流 ID，3- 本地输入源，4- 虚拟屏服务器通道，5-拼接通道，6- 屏幕服务器

byStreamId

流 ID，通道类型 *byChanType* 为 2 时有效

byRes1

保留，置为 0

dwChannel

通道号，通道类型 *byChanType* 为 0、1、3、4、5、6 时有效

byRes2

保留，置为 0

byAddress

设备域名

wDVRPort

端口号

byChannel

该参数无效，通道号见 dwChannel，建议和 dwChannel 设置相同的值

byTransProtocol

传输协议类型：0- TCP，1- UDP

byTransMode

传输码流模式：0- 主码流，1- 子码流

byFactoryType

前端设备厂家类型，通过接口 [NET_DVR_GetIPProtoList_V41](#) 获取

byDeviceType

设备类型(视频综合平台智能板使用)：1- 解码器（此时根据视频综合平台能力集中

byVcaSupportChanMode 字段来决定是使用解码通道还是显示通道），2- 编码器，3- X86 高清输入

byDispChan

显示通道号，智能配置使用

bySubDispChan

显示通道子通道号，智能配置时使用

byResolution

1- CIF，2- 4CIF，3- 720P，4- 1080P，5- 500w，大屏控制器使用，根据该参数分配解码资源

byRes

保留，置为 0

sUserName

监控主机登陆帐号

sPassword

监控主机密码

8.31 NET_DVR_DEV_CHAN_INFO_V41:前端编码设备信息

```
struct{
    BYTE    byAddress[MAX_DOMAIN_NAME];
    WORD    wDVRPort;
    BYTE    byChannel;
    BYTE    byTransProtocol;
    BYTE    byTransMode;
    BYTE    byFactoryType;
    BYTE    byDeviceType;
    BYTE    byDispChan;
    BYTE    bySubDispChan;
    BYTE    byResolution;
    BYTE    byRes[2];
    BYTE    sUserName[NAME_LEN];
    BYTE    sPassword[PASSWD_LEN];
}NET_DVR_DEV_CHAN_INFO_V41,*LPNET_DVR_DEV_CHAN_INFO_V41;
```

Members

byAddress

设备 IP 地址或者域名

wDVRPort

设备端口号

byChannel

通道号，目前设备模拟通道号是从 1 开始的，对于 9000 等设备的 IPC 接入，数字通道号从 33 开始

byTransProtocol

传输协议类型：0-TCP，1-UDP

byTransMode

传输码流模式：0- 主码流，1- 子码流

byFactoryType

前端设备厂家类型，通过接口 [NET_DVR_GetIPCProtoList_V41](#) 获取

byDeviceType

设备类型(视频综合平台智能板使用)：1- IPC，2- ENCODER

byDispChan

显示通道号（智能配置使用），根据能力集决定使用解码通道还是显示通道

bySubDispChan

显示通道子通道号（智能配置时使用）

byResolution

分辨率：1- CIF，2- 4CIF，3- 720P，4- 1080P，5- 500W

byRes

保留，置为 0

sUserName

设备登陆帐号

sPassword

设备密码

8.32 NET_DVR_DEV_DDNS_INFO:动态域名参数配置

```
struct{
    BYTE    byDevAddress[MAX_DOMAIN_NAME];
    BYTE    byTransProtocol;
    BYTE    byTransMode;
    BYTE    byDdnsType;
    BYTE    byRes1;
    BYTE    byDdnsAddress[MAX_DOMAIN_NAME];
    WORD    wDdnsPort;
    BYTE    byChanType;
    BYTE    byFactoryType;
    DWORD   dwChannel;
    BYTE    byStreamId[STREAM_ID_LEN];
    BYTE    sUserName[NAME_LEN];
    BYTE    sPassword[PASSWD_LEN];
    WORD    wDevPort;
    BYTE    byRes2[2];
}NET_DVR_DEV_DDNS_INFO,*LPNET_DVR_DEV_DDNS_INFO;
```


Members*byDevAddress*

设备域名(IPServer 或 hiDDNS 时可填设备序列号或者别名)

byTransProtocol

传输协议类型: 0- TCP, 1- UDP, 2- 多播

byTransMode

传输码流模式: 0- 主码流, 1- 子码流

byDdnsType

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

byRes1

保留

byDdnsAddress

DDNS 服务器地址

wDdnsPort

DDNS 服务器端口号

byChanType

通道类型: 0-普通通道, 1-零通道, 2-流 ID

*byFactoryType*前端设备厂家类型, 通过接口 [NET_DVR_GetIPCProtoList_V41](#) 获取*dwChannel*

设备通道号

byStreamId

流 ID

sUserName

设备登陆帐号

sPassword

设备密码

wDVRPort

设备端口号

byRes2

保留

8.33 NET_DVR_DEVICE_BOARD_EXCEPINFO:子板异常信息

```

struct{
    DWORD    dwSize;
    BYTE     byExceptNum;
    BYTE     byRes1[3];
    BYTE     byMajor[MAX_SINGLE_BOARD_EXCEPTION_NUM];
    WORD     wMinor[MAX_SINGLE_BOARD_EXCEPTION_NUM];
    BYTE     byRes2[32];
}NET_DVR_DEVICE_BOARD_EXCEPINFO, *LPNET_DVR_DEVICE_BOARD_EXCEPINFO;

```

Members*dwSize*

结构体大小

byExceptNum

异常项数

byRes1

保留，置为 0

byMajor

主类型，每位数组表示一种异常主类型，主类型取值定义如下表所示：

byMajor 宏定义	宏定义值	含义
EXCEPTION_MAJOR_MAINBOARD_BOOT	0x1	主板启动异常
EXCEPTION_MAJOR_SUBBOARD_BOOT	0x2	子板启动异常
EXCEPTION_MAJOR_SUBBOARD_HARDWARE	0x3	子板硬件异常
EXCEPTION_MAJOR_FPGA	0x4	FPGA 异常
EXCEPTION_MAJOR_DSP	0x5	DSP 异常
EXCEPTION_MAJOR_ARM	0x6	ARM 异常
EXCEPTION_MAJOR_BACKBOARD	0x7	背板异常
EXCEPTION_MAJOR_SUBBOARD_NET	0x8	子板网络异常

wMinor

次类型，保留

byRes2

保留，置为 0

8.34 NET_DVR_DEVICE_RUN_STATUS: 设备运行状态

struct{

DWORD dwSize;

DWORD dwMemoryTotal;

DWORD dwMemoryUsage;

BYTE byCPUUsage;

BYTE byMainFrameTemp;

BYTE byBackPanelTemp;

BYTE byRes1;

BYTE byLeftDecResource[32];

BYTE byRes[92];

}NET_DVR_DEVICE_RUN_STATUS, *LPNET_DVR_DEVICE_RUN_STATUS;

Members

dwSize

结构体大小

dwMemoryTotal

内存总量，单位：Kbyte

dwMemoryUsage

内存使用量，单位：Kbyte

byCPUUsage

CPU 使用率，0~100

byMainFrameTemp

机箱温度，单位：摄氏度

byBackPanelTemp

背板温度，单位：摄氏度

byRes1

保留，置为 0

byLeftDecResource

各解码板剩余解码资源，以解 D1 分辨率资源为单位，byLeftDecResource[i]表示槽位号 i+1 的解码板剩余资源，0xff 表示无效（非解码板或没插板子）

byRes

保留，置为 0

8.35 NET_DVR_DEVICE_SUBBOARD_INFO:设备子板信息

```
struct{
    DWORD                dwSize;
    BYTE                 byBackBoardType;
    BYTE                 bySoltNum;
    BYTE                 byBoardNum;
    BYTE                 byRes1;
    NET_DVR_SUBBOARD_INFO struSubBoadInfo[MAX_SUBBOARD_NUM];
    BYTE                 byRes2[32];
}NET_DVR_DEVICE_SUBBOARD_INFO, *LPNET_DVR_DEVICE_SUBBOARD_INFO;
```

Members

dwSize

结构体大小

byBackBoardType

背板类型：1- 4U，2- 8U，3- 13U，0xff- 未知类型

bySoltNum

槽位数目

byBoardNum

有效子板数目

byRes1

保留，置为 0

struSubBoadInfo

子板信息，数组下标跟子板一一对应，前 byBoardNum 项有效

byRes2

保留，置为 0

8.36 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

byIPCPlug

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。设备类型定义如下所示：

宏定义	宏定义值	设备类型
DS_AIO_M_CLASS	353	视频综合平台一体机

DS_iVMSE_AIO_CLASS	352	一体机 x86 服务器子系统
DS_6402_AIO_CLASS	351	一体机 Netra 编码器
DS_6504D_AIO_CLASS	349	一体机 Netra 高清解码器
DS64XXHD_T	701	64-T 高清解码器
DS_64XXHD_S	706	DS-64xxHD-S 高清解码器
DS_C10H	161	多屏控制器
DS_C10N_SERVER	166	分布式服务器

8.37 NET_DVR_DEVICEINFO_V40:设备参数

```

struct{
    NET_DVR_DEVICEINFO_V30    struDeviceV30;
    BYTE                      bySupportLock;
    BYTE                      byRetryLoginTime;
    BYTE                      byPasswordLevel;
    BYTE                      byRes1;
    DWORD                    dwSurplusLockTime;
    BYTE                      byCharEncodeType;
    BYTE                      byRes2[255];
}NET_DVR_DEVICEINFO_V40,*LPNET_DVR_DEVICEINFO_V40;

```

Members

struDeviceV30

设备参数

bySupportLock

设备是否支持锁定功能，bySupportLock 为 1 时，dwSurplusLockTime 和 byRetryLoginTime 有效

byRetryLoginTime

剩余可尝试登陆的次数，用户名、密码错误时，此参数有效

byPasswordLevel

密码安全等级：0- 无效，1- 默认密码，2- 有效密码，3- 风险较高的密码，当管理员用户的密码为出厂默认密码（12345）或者风险较高的密码时，建议上层客户端提示用户更改密码

byRes1

保留，置为 0

dwSurplusLockTime

剩余时间，单位：秒，用户锁定时此参数有效。在锁定期间，用户尝试登陆，不管用户名密码输入对错，设备锁定剩余时间重新恢复到 30 分钟

byCharEncodeType

字符编码类型（SDK 所有接口返回的字符串编码类型，透传接口除外）：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)

byRes2

保留，置为 0

Remarks

将密码输入分为数字(0~9)、小写字母(a~z)、大写字母(A~Z)、特殊符号 (:\"除外) 4 类, 等级分为 4 个等级, 如下所示:

- 等级 0 (风险密码): 密码长度小于 8 位, 或者只包含 4 类字符中的任意一类, 或者密码与用户名一样, 或者密码是用户名的倒写。例如: 12345、abcdef。
- 等级 1 (弱密码): 包含两类字符, 且组合为 (数字+小写字母) 或 (数字+大写字母), 且长度大于等于 8 位。例如: abc12345、123ABCDEF。
- 等级 2 (中密码): 包含两类字符, 且组合不能为 (数字+小写字母) 和 (数字+大写字母), 且长度大于等于 8 位。例如: 12345***++、ABCDabcd。
- 等级 3 (强密码): 包含三类字符及以上, 且长度大于等于 8 位。例如: Abc12345、abc12345++。

8.38 NET_DVR_DEVICEID_INFO:设备 ID 信息

```
struct{
    DWORD    dwSize;
    DWORD    dwDeviceIndex;
    BYTE     byWallNo;
    BYTE     byRes1[27];
    DWORD    dwChan;
    DWORD    dwInputSignalIndex;
    BYTE     byRes2[60];
}NET_DVR_DEVICEID_INFO,*LPNET_DVR_DEVICEID_INFO;
```

Members

dwSize

结构体大小

dwDeviceIndex

设备序号/槽位号, 集中式通过槽位号配置解码板信息, 分布式通过设备序号配置

byWallNo

墙号, 从 1 开始。暂不使用, 保留

byRes1

保留

dwChan

通道号, BNC 处理器: 1~16, RGB 处理器为 1, 转码处理器: 1~4

dwInputSignalIndex

信号源索引, 保留

byRes2

保留

8.39 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。

8.40 NET_DVR_DEVLIST:分布式子设备信息列表

```
struct{
    DWORD    dwSize;
    DWORD    dwDevNums;
    BYTE     *pBuffer;
    BYTE     byRes1[3];
    DWORD    dwBufLen;
    BYTE     byRes2[64];
}NET_DVR_DEVLIST,*LPNET_DVR_DEVLIST;
```

Members

dwSize

结构体大小

dwDevNums

设备返回的分布式子设备数量

pBuffer

子设备信息列表，dwDevNums*[NET_DVR_SINGLE_DEV_INFO](#)

byRes1

保留

dwBufLen

所分配指针长度，输入参数

byRes2

保留

8.41 NET_DVR_DIFFDEV_INFO:子设备信息联合体

```
union{
    NET\_DVR\_CODEDEV\_INFO      struCodeDevInfo;
    NET\_DVR\_DECODEDEV\_INFO    struDecodeDevInfo;
    NET\_DVR\_NETSIGNALDEV\_INFO struNetSignalInfo;
```

```
}NET_DVR_DIFFDEV_INFO,*LPNET_DVR_DIFFDEV_INFO;
```

Members

struCodeDevInfo

编码设备信息

struDecodeDevInfo

解码设备信息

struNetSignalInfo

网络信号源信息

Remarks

- 网络信号源在系统中层次应该是在码流处理器的下层，负责将网络码流传给码流处理器,但是把结构体放在这里跟码流处理和其他设备同一级。
- 实际使用中，如果是本地信号，客户端上直接拖显示处理器的通道到一个窗口上。而网络信号需要拖动网络信号源的通道到一个窗口上。

8.42 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

strulP
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

8.43 NET_DVR_DISPLAY_COLOR_CTRL:显示单元颜色控制参数

```
struct{
    BYTE    byColorType;
    BYTE    byScale;
    BYTE    byRes[14];
}NET_DVR_DISPLAY_COLOR_CTRL,*LPNET_DVR_DISPLAY_COLOR_CTRL;
```

Members

byColorType
类型：1- 亮度，2- 对比度，3- 饱和度，4- 清晰度

byScale
控制取值：-1 或者+1

byRes
保留

8.44 NET_DVR_DISPLAY_POSITION_CTRL:显示单元位置控制参数

```
struct{
    BYTE    byPositionType;
    BYTE    byScale;
    BYTE    byRes[14];
}NET_DVR_DISPLAY_POSITION_CTRL,*LPNET_DVR_DISPLAY_POSITION_CTRL;
```

Members

byPositionType

类型: 1- 水平位置, 2- 垂直位置

byScale

控制取值: -1 或者+1

byRes

保留

8.45 NET_DVR_DISPLAYCFG:显示输出参数配置

```
struct {
    DWORD    dwSize;
    NET\_DVR\_DISPLAYPARAM    struDisplayParam[MAX_DISPLAY_NUM] ;
    BYTE    byRes[128];
}NET_DVR_DISPLAYCFG,*LPNET_DVR_DISPLAYCFG;
```

Members

dwSize

结构体大小

struDisplayParam

显示输出参数

byRes

保留

8.46 NET_DVR_DISPLAYPARAM:显示输出参数

```
struct {
    DWORD    dwDisplayNo;
    BYTE    byDispChanType;
    BYTE    byRes[11];
}NET_DVR_DISPLAYPARAM,*LPNET_DVR_DISPLAYPARAM;
```

Members

dwDisplayNo

显示输出号(组合)

byDispChanType

输出口类型: 1- BNC, 2- VGA, 3- HDMI, 4- DVI, 5- SDI, 6- FIBER, 7- RGB, 8- YPrPb, 9- VGA/HDMI/DVI

自适应, 10- 3GSDI, 11- VGA/DVI 自适应, 12- HDBaseT, 0xff- 无效

byRes

保留

8.47 NET_DVR_DISPINPUT_CFG:显示输入参数

```
struct{
    DWORD    dwSize;
    DWORD    dwDisplInputNo;
    DWORD    dwEDIDFileNo;
    BYTE     byRes[32];
}NET_DVR_DISPINPUT_CFG, *LPNET_DVR_DISPINPUT_CFG;
```

Members

dwSize

结构体大小

dwDisplInputNo

输入口号(组合)

dwEDIDFileNo

关联的 EDID 文件编号

byRes

保留 , 置为 0

8.48 NET_DVR_DISPINPUT_CFG_LIST:显示输入参数列表

```
struct{
    DWORD                dwSize;
    DWORD                dwDisplInputNum;
    LPNET\_DVR\_DISPINPUT\_CFG lpstruBuffer;
    DWORD                dwBufferSize;
    BYTE                 byRes[32];
}NET_DVR_DISPINPUT_CFG_LIST, *LPNET_DVR_DISPINPUT_CFG_LIST;
```

Members

dwSize

结构体大小

dwDisplInputNum

输入口号个数, 对应 lpstruBuffer 中返回的结构个数

lpstruBuffer

缓冲区指针, 指向 dwDisplInputNum 个 NET_DVR_DISPINPUT_CFG 结构体数据

dwBufferSize

缓冲区大小

byRes

保留, 置为 0

Remarks

- 获取列表时，可以先分配一块足够大的缓冲区 `lpstruBuffer`，然后调用一次 `NET_DVR_GetSTDConfig` 接口，即可将设备发来的数据写入缓冲区；或者先将缓冲区指针 `lpstruBuffer` 置为 `NULL`，第一次调用 `NET_DVR_GetSTDConfig` 接口时返回 `NET_DVR_DISPINPUT_CFG` 结构的个数（`dwDispInputNum`），然后根据 `dwDispInputNum` 的值申请相应的缓冲区，再次调用 `NET_DVR_GetSTDConfig` 接口把设备发来的数据写入缓冲区。
- 设置绑定时，若 EDID 文件编号为 0 则表示取消当前显示输入和 EDID 文件的绑定。

8.49 NET_DVR_DVCS_STATE_ALARM:分布式多屏控制器报警信息

```
struct{
    DWORD    dwSize;
    BYTE     byAlarmType;
    BYTE     byDeviceType;
    BYTE     byWallNo;
    BYTE     byDeviceChanIndex;
    DWORD    dwDeviceIndex;
    WORD     wResolutionX;
    WORD     wResolutionY;
    WORD     wTemperature;
    BYTE     byRes[86];
}NET_DVR_DVCS_STATE_ALARM,*LPNET_DVR_DVCS_STATE_ALARM;
```

Members

dwSize

结构体大小

byAlarmType

报警类型：1- 设备上线，2- 设备离线，3- 服务器上线，4- 服务器离线，5-管理员登陆，6-预案开始，7-预案结束，8-分辨率变化，9-图像切割失败，10-设备资源不足，11-温度异常

byDeviceType

设备类型(只能获取)，取值定义如下：

```
enum _NET_DVCS_DEVICE_TYPE_{
    NET_DVR_BNC_PROC = 1,
    NET_DVR_RGB_PROC,
    NET_DVR_STP_PROC,
    NET_DVR_DISP_PROC,
    NET_DVR_NETSIGNAL
}NET_DVCS_DEVICE_TYPE
```

NET_DVR_BNC_PROC

BNC 处理器

NET_DVR_RGB_PROC

RGB 处理器

NET_DVR_STP_PROC

转码处理器

NET_DVR_DISP_PROC

显示处理器

NET_DVR_NETSIGNAL

网络信号源

byWallNo

墙号，从 1 开始，0 表示无效

byDeviceChanIndex

设备通道号，0 表示无效，其他值表示通道号

dwDeviceIndex

设备序号，0 表示无效

wResolutionX

分辨率的宽度，单位：像素，0 表示无效

wResolutionY

分辨率的高度，单位：像素，0 表示无效

wTemperature

设备温度，单位：摄氏度，0 表示无效

byRes

保留，置为 0

Remarks

分辨率的变化，可能导致该分辨率下图像不能被切割并放大到全屏。前端输入分辨率的变化，也可能导致解码资源不足，在解码资源不足的情况下，该解码设备的所有通道都会受影响，与通道号无关。

8.50 NET_DVR_EDID_FILE_INFO:EDID 配置文件信息

```
struct{
    DWORD    dwSize;
    DWORD    dwEDIDFileNo;
    BYTE     byFileName[FILE_NAME_LEN];
    BYTE     byRes[32];
}NET_DVR_EDID_FILE_INFO, *LPNET_DVR_EDID_FILE_INFO;
```

Members*dwSize*

结构体大小

dwEDIDFileNo

EDID 文件编号，从 1 开始

byFileName

EDID 文件名称

byRes

保留，置为 0

8.51 NET_DVR_EDID_FILE_INFO_LIST:EDID 配置文件信息列表

```
struct{
    DWORD    dwSize;
    DWORD    dwEDIDFileNum;
```

```

LPNET_DVR_EDID_FILE_INFO  lpstruBuffer;
DWORD                      dwBufferSize;
BYTE                      byRes[32];
}NET_DVR_EDID_FILE_INFO_LIST, *LPNET_DVR_EDID_FILE_INFO_LIST;

```

Members

dwSize

结构体大小

dwEDIDFileNum

EDID 配置文件个数

lpstruBuffer

EDID 配置文件信息数据缓冲区指针，指向 dwEDIDFileNum 个 NET_DVR_EDID_FILE_INFO 结构体

dwBufferSize

缓冲区大小，dwEDIDFileNum*sizeof(NET_DVR_EDID_FILE_INFO)

byRes

保留，置为 0

Remarks

获取列表时，可以先分配一块足够大的缓冲区 lpstruBuffer，然后调用一次 [NET_DVR_GetSTDConfig](#) 接口直接获取所有文件信息；或者先将缓冲区指针 lpstruBuffer 置为 NULL，第一次调用 [NET_DVR_GetSTDConfig](#) 接口时返回 NET_DVR_EDID_FILE_INFO 结构的个数 dwEDIDFileNum，然后根据 dwEDIDFileNum 的值申请相应的缓冲区，再次调用 [NET_DVR_GetSTDConfig](#) 接口获取文件信息列表。

8.52 NET_DVR_ETHERNET_CHANNEL_INFO:端口聚合参数

```

struct{
    DWORD    dwSize;
    BYTE     byConverge[MAX_ETHERNET_PORT_NUM];
    BYTE     byRes[32];
}NET_DVR_ETHERNET_CHANNEL_INFO,*LPNET_DVR_ETHERNET_CHANNEL_INFO;

```

Members

dwSize

结构体大小

byConverge

byConverge[i] = j 表示将第 i 个端口加入到第 j 条链路，j 的取值为 1~8，j=0 表示移除

byRes

保留

8.53 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。

8.54 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

8.55 NET_DVR_EXTERNAL_MATRIX_CFG:矩阵配置信息

```
struct{
    DWORD          dwSize;
    BYTE           byValid;
    BYTE           byRes1[3];
    BYTE           sMatrixName[NAME_LEN];
    DWORD          dwMatrixID;
    WORD           wMatrixInputChanNum;
    WORD           wMatrixOutputChanNum;
    WORD           wMatrixOutputChanRef[MAX_CAM_COUNT];
    BYTE           byMatrixChanType;
    BYTE           byMatrixProtocol;
    BYTE           byMatrixType;
    BYTE           byRes2;
    NET\_MATRIX\_UNION struMatrixUnion;
    BYTE           byRes3[128];
}NET_DVR_EXTERNAL_MATRIX_CFG,*LPNET_DVR_EXTERNAL_MATRIX_CFG;
```

Members

dwSize

结构体大小

byValid

是否有效：0-无效，1-有效，有效时为设置矩阵，无效时为删除矩阵

byRes1

保留，置为 0

sMatrixName

矩阵的名称

dwMatrixID

矩阵的 ID 号，获取所有矩阵配置时使用该字段，其他接口调用时该值无效

wMatrixInputChanNum

矩阵的输入通道数目

wMatrixOutputChanNum

矩阵的输出通道数目

wMatrixOutputChanRef

设备输入和矩阵输出的绑定关系，数组下标表示控制器设备输入通道号，元素的值表示矩阵的输出通道号，元素值大于 0 表示绑定矩阵输出通道，元素值为 0 时表示不绑定

byMatrixChanType

矩阵的通道类型：1-BNC，2-VGA，3-RGB，4-DVI

byMatrixProtocol

矩阵所用协议：1-ZT1.0，2- ZT2.0，3- Extron，4- Creator

byMatrixType

矩阵类型：1-模拟矩阵，2-数字矩阵，集中式多屏控制器 V2.1 只支持模拟矩阵

byRes2

保留，置为 0

struMatrixUnion

模拟或者数字矩阵参数联合体

byRes3

保留，置为 0

8.56 NET_DVR_FUSION_CFG:图像融合配置

```
struct{
    DWORD    dwSize;
    BYTE     byFusion;
    BYTE     byUseHistoryMap;
    BYTE     byRes[30];
}NET_DVR_FUSION_CFG, *LPNET_DVR_FUSION_CFG;
```

Members

dwSize

结构体大小

byFusion

是否融合：0- 未融合或取消融合，非 0- 融合

byUseHistoryMap

是否使用历史融合映射表：0- 不使用，非 0- 使用

byRes

保留，置为 0

Remarks

图像融合，即将多个 IPC 的图像经过一系列的算法拼接成一个全景图像，拼接融合过程需要一定的时间。如果 IPC 的位置没有更改过，先前配置过同样的拼接模式，使用历史融合映射表，可以缩短融合消耗的时间。

设备是否支持历史融合映射表，可以通过设备能力集进行判断，对应相机拼接能力集

([CamFusionAbility](#))，相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：CAM_FUSION_ABILITY，节点：[<supportHistoryMap>](#)。

8.57 NET_DVR_FUSION_SCALE:图像融合规模参数配置

```
struct{
    DWORD    dwSize;
    WORD     wWidth;
    WORD     wHeight;
    BYTE     byRes[32];
```

```
}NET_DVR_FUSION_SCALE, *LPNET_DVR_FUSION_SCALE;
```

Members

dwSize

结构体大小

wWidth

图像融合宽(横向图像数目)

wHeight

图像融合高(纵向图像数目)

byRes

保留，置为 0

Remarks

设备是否支持图像融合能力或者支持的参数能力，可以通过设备能力集进行判断，对应相机拼接能力集（[CamFusionAbility](#)），相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：CAM_FUSION_ABILITY，节点：<FusionScale>。

8.58 NET_DVR_GBT28181_ACCESS_CFG:GB28181 协议接入配置

```
struct{
    DWORD    dwSize;
    BYTE     byEnable;
    BYTE     byTransProtocol;
    WORD     wLocalSipPort;
    char     szServerID[MAX_SERVERID_LEN];
    char     szServerDomain[MAX_SIPSERVER_ADDRESS_LEN];
    char     szSipServerAddress[MAX_SIPSERVER_ADDRESS_LEN];
    WORD     wServerSipPort;
    BYTE     byRes3[2];
    char     szSipUserName[MAX_SERVERNAME_LEN];
    char     szSipAuthenticateID[MAX_AUTHENTICATEID_LEN];
    char     szSipAuthenticatePasswd[MAX_AUTHENTICATEPASSWD_LEN];
    DWORD    dwRegisterValid;
    BYTE     byHeartbeatInterval;
    BYTE     byMaxHeartbeatTimeOut;
    BYTE     byStreamType;
    BYTE     byDeviceStatus;
    DWORD    dwRegisterInterval;
    DWORD    dwAutoAllocChannelID;
    char     szDeviceDomain[MAX_SIPSERVER_ADDRESS_LEN];
    BYTE     byRes4[116];
}NET_DVR_GBT28181_ACCESS_CFG,*LPNET_DVR_GBT28181_ACCESS_CFG;
```

Members

dwSize

结构体大小

byEnable

28181 协议使能：0- 关闭，1- 开启

byTransProtocol

传输协议：0- UDP(默认)，1- TCP

wLocalSipPort

设备 SIP 端口，取值范围：1024~65535

szServerID

服务器 ID，64 字节字符串，仅限数字，由 GB28181 服务器提供

szServerDomain

服务器域

szSipServerAddress

SIP 服务器地址，支持域名和 IP 地址

wServerSipPort

服务器 SIP 端口：1024~65535

byRes3

保留，置为 0

szSipUserName

SIP 用户名称，64 字节字符串(第三方 SIP 终端请求 IPC 服务时使用)

szSipAuthenticateID

SIP 用户认证 ID(注册用户名)，64 字节字符串

szSipAuthenticatePasswd

SIP 用户认证密码，32 字节字符串(IPC 设备注册到 SIP 服务器时使用)

dwRegisterValid

注册有效期，单位：秒，默认：3600

byHeartbeatInterval

心跳间隔，单位：秒，默认：10 秒

byMaxHeartbeatTimeOut

最大心跳超时次数，默认：3 次

byStreamType

取流类型：0- 主码流，1- 子码流，2- 三码流

byDeviceStatus

设备是否在线状态：0- 保留，1- 在线，2- 离线

dwRegisterInterval

注册间隔（注册失败后再次注册的时间间隔），取值范围：60~600s，默认：60s

dwAutoAllocChannelID

是否自动分配通道 ID，按位表示，取值：0 为手动配置，1 为自动分配

bit1- 自动分配编码通道 ID，

bit2- 自动分配报警输入通道，

bit3- 自动分配解码通道 ID

szDeviceDomain

设备域

byRes4

保留，置为 0

Remarks

GB28181 详细内容请参考《安全防范视频监控联网系统信息传输、交换、控制技术要求》（即 GB28181

协议文档)。

8.59 NET_DVR_GBT28181_CHANINFO_CFG:GB28181 协议设备的编码通道配置

```
struct{
    DWORD    dwSize;
    char      szVideoChannelNumID[MAX_COMPRESSIONID_LEN];
    BYTE      byRes[256];
}NET_DVR_GBT28181_CHANINFO_CFG,*LPNET_DVR_GBT28181_CHANINFO_CFG;
```

Members

dwSize

结构体大小

szVideoChannelNumID

设备视频通道编码 ID，64 字节字符串，仅限数字

byRes

保留，置为 0

See Also

GB28181 包括编码 ID 规定，详细内容请参考《安全防范视频监控联网系统信息传输、交换、控制技术 要求》(即 GB28181 协议文档)。

8.60 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

8.61 NET_DVR_IN_PARAM:输入参数

```
struct{
    NET\_DVR\_BUF\_INFO    struCondBuf;
    NET\_DVR\_BUF\_INFO    struInParamBuf;
    DWORD               dwRecvTimeout;
    BYTE                byRes[32];
}NET_DVR_IN_PARAM,LPNET_DVR_IN_PARAM;
```

Members

struCondBuf

条件缓冲区

struInParamBuf

参数缓冲区

dwRecvTimeout

接收数据超时时间，单位：ms，为 0 时采用接口默认超时

byRes

保留，置为 0

8.62 NET_DVR_INPUT_BOARD_CFG:输入板参数配置

```
struct{
    DWORD    dwSize;
    DWORD    dwSlotNo;
    BYTE     byFullFrameEnable;
    BYTE     byRes[3];
    BYTE     byRes1[64];
}NET_DVR_INPUT_BOARD_CFG, *LPNET_DVR_INPUT_BOARD_CFG; //输入板参数配置
```

Members

dwSize

结构体大小

dwSlotNo

输入板槽位号

byFullFrameEnable

是否开启全帧率畅显：0- 不开启，1- 开启

byRes

保留，置为 0

byRes1

保留，置为 0

Remarks

设备是否支持输入板配置能力，可以通过设备能力集进行判断，对应电视墙能力集([WallAbility](#))，相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：WALL_ABILITY，节点：<InputBoardCfgList>。

8.63 NET_DVR_INPUT_BOARD_CFG_LIST:输入板配置信息列表

```
struct{
    DWORD                dwSize;
    NET\_DVR\_INPUT\_BOARD\_CFG struBoardList[MAX_NUM_INPUT_BOARD];
}NET_DVR_INPUT_BOARD_CFG_LIST, *LPNET_DVR_INPUT_BOARD_CFG_LIST;
```

Members

dwSize

结构体大小

struBoardList

输入板配置信息列表

Remarks

设备是否支持输入板配置能力，可以通过设备能力集进行判断，对应电视墙能力集([WallAbility](#))，相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：WALL_ABILITY，节点：<InputBoardCfgList>。

8.64 NET_DVR_INPUT_INTERFACE_CTRL:输入接口控制

```
struct{
    BYTE    byInputSourceType;
    BYTE    byRes[15];
}NET_DVR_INPUT_INTERFACE_CTRL, *LPNET_DVR_INPUT_INTERFACE_CTRL;
```

Members

byInputSourceType

输入源类型

```
enum _INPUT_INTERFACE_TYPE_{
    INTERFACE_VGA    = 0,
    INTERFACE_SVIDEO,
    INTERFACE_YPBPR,
    INTERFACE_DVI,
    INTERFACE_BNC,
    INTERFACE_DVI_LOOP,
    INTERFACE_BNC_LOOP,
    INTERFACE_HDMI
}INPUT_INTERFACE_TYPE
```

INTERFACE_VGA

VGA 输入

INTERFACE_SVIDEO

S-Video 输入

INTERFACE_YPBPR

YPbPr 输入

INTERFACE_DVI

DVI 输入

INTERFACE_BNC

BNC 输入
INTERFACE_DVI_LOOP
 DVI（环通）输入
INTERFACE_BNC_LOOP
 BNC（环通）输入
INTERFACE_HDMI
 HDMI 输入

byRes

保留

8.65 **NET_DVR_INPUT_SIGNAL_LIST**: 输入信号源列表

```

struct{
    DWORD    dwSize;
    DWORD    dwInputSignalNums;
    BYTE     *pBuffer;
    BYTE     byRes1[3];
    DWORD    dwBufLen;
    BYTE     byRes2[64];
}NET_DVR_INPUT_SIGNAL_LIST,*LPNET_DVR_INPUT_SIGNAL_LIST;
  
```

Members

dwSize

结构体大小

dwInputSignalNums

设备输入信号源数量

pBuffer

输入信号源，指向 *dwInputSignalNums* 个 [NET_DVR_INPUTSTREAMCFG_V40](#) 结构体大小的缓冲区

byRes1

保留，置为 0

dwBufLen

所分配缓冲区长度

byRes2

保留，置为 0

Remarks

1. 调用 [NET_DVR_GetInputSignalList_V40](#)，将该结构体中的 *pBuffer* 置为空且 *dwBufLen* 为 0，此时获取输入信号源个数（*dwInputSignalNums*）；
2. 根据接口返回的数量分配 *pBuffer* 的大小，然后再次调用 [NET_DVR_GetInputSignalList_V40](#)，此时获取输入信号源列表（*pBuffer*）。

8.66 NET_DVR_INPUTSTREAM_PTZCFG_PARAM:输入源反向云台控制配

置参数

```
struct{
    DWORD    dwSize;
    WORD     wPTZProtocol;
    BYTE     byRes[34];
}NET_DVR_INPUTSTREAM_PTZCFG_PARAM, *LPNET_DVR_INPUTSTREAM_PTZCFG_PARAM;
```

Members

dwSize

结构体大小

wPTZProtocol

控球协议类型，索引值，设备支持的云台控制协议可以通过 [NET_DVR_GetPTZProtocol_Ex](#) 获取

byRes

保留，置为 0

8.67 NET_DVR_INPUTSTREAMCFG:输入源参数

```
struct{
    DWORD                dwSize;
    BYTE                 byValid;
    BYTE                 byCamMode;
    WORD                 wInputNo;
    BYTE                 sCamName[NAME_LEN];
    NET\_DVR\_VIDEOEFFECT struVideoEffect;
    NET\_DVR\_PU\_STREAM\_CFG struPuStream;
    WORD                 wBoardNum;
    WORD                 wInputIdxOnBoard;
    DWORD                dwResolution;
    BYTE                 byVideoFormat;
    BYTE                 byStatus;
    BYTE                 sGroupName[NAME_LEN];
    BYTE                 byJointMatrix;
    BYTE                 byRes;
}NET_DVR_INPUTSTREAMCFG, *LPNET_DVR_INPUTSTREAMCFG;
```

Members

dwSize

结构体大小

byValid

是否有效：0- 否，1- 是

byCamMode

信号输入源类型

wInputNo

信号源序号

sCamName

信号输入源名称

struVideoEffect

视频参数

struPuStream

输入流参数，ip 输入时使用

wBoardNum

信号源所在的板卡号，只能获取

wInputIdxOnBoard

信号源在板卡上的位置，只能获取

dwResolution

分辨率

byVideoFormat

视频制式：0- 无，1- NTSC，2- PAL

byStatus

信号源状态：0- 字段无效，1- 有信号，2- 无信号，3- 异常

sGroupName

网络信号源分组的组名

byJointMatrix

关联矩阵：0-不关联，1-关联，当输入信号源为 NET_DVR_CAM_BNC、NET_DVR_CAM_VGA、NET_DVR_CAM_DVI、NET_DVR_CAM_HDMI 中的一种时，该参数有效。

byRes

保留，置为 0

Remarks

对于本地信号源和矩阵信号源，只有 *sCamName* 和 *struVideoEffect* 可以设置；其他参数只能获取，不能设置。

8.68 NET_DVR_INPUTSTREAMCFG_V40:输入源参数

```

struct{
    DWORD          dwSize;
    BYTE           byValid;
    BYTE           byCamMode;
    WORD           wInputNo;
    BYTE           sCamName[NAME_LEN];
    NET\_DVR\_VIDEOEFFECT struVideoEffect;
    NET\_DVR\_PU\_STREAM\_CFG struPuStream;
    WORD           wBoardNum;
    WORD           wInputIdxOnBoard;
    DWORD          dwResolution;
    BYTE           byVideoFormat;
    BYTE           byStatus;

```

```

BYTE                sGroupName[NAME_LEN];
BYTE                byJointMatrix;
BYTE                byJointNo;
BYTE                byColorMode;
BYTE                byScreenServer;
BYTE                byRes1[2];
DWORD               dwInputSignalNo;
BYTE                byRes[120];
}NET_DVR_INPUTSTREAMCFG_V40,*LPNET_DVR_INPUTSTREAMCFG_V40;

```

Members

dwSize

结构体大小

byValid

是否有效：0- 否，1- 是

byCamMode

信号输入源类型：

```

enum _NET_DVR_CAM_MODE_{
    NET_DVR_UNKNOW    = 0,
    NET_DVR_CAM_BNC,
    NET_DVR_CAM_VGA,
    NET_DVR_CAM_DVI,
    NET_DVR_CAM_HDMI,
    NET_DVR_CAM_IP,
    NET_DVR_CAM_RGB,
    NET_DVR_CAM_DECODER,
    NET_DVR_CAM_MATRIX,
    NET_DVR_CAM_YPBPR,
    NET_DVR_CAM_USB,
    NET_DVR_CAM_SDI,
    NET_DVR_CAM_HDI,
    NET_DVR_CAM_DP,
    NET_DVR_CAM_HDTV,
    NET_DVR_CAM_JOINT,
    NET_DVR_CAM_HDBASET,
    NET_DVR_CAM_DVIT,
    NET_DVR_CAM_FUSION,
    NET_DVR_CAM_VSCREEN,
    NET_DVR_CAM_FIBER,
    NET_DVR_CAM_3GSDI
}NET_DVR_CAM_MODE

```

NET_DVR_UNKNOW

无效

NET_DVR_CAM_BNC

BNC 输入

NET_DVR_CAM_VGA

VGA 输入

NET_DVR_CAM_DVI

DVI 输入

NET_DVR_CAM_HDMI

HDMI 输入

NET_DVR_CAM_IP

IP 输入

NET_DVR_CAM_RGB

RGB 输入

NET_DVR_CAM_DECODER

解码板输入

NET_DVR_CAM_MATRIX

矩阵信号源

NET_DVR_CAM_YPBPR

YPbPr 输入

NET_DVR_CAM_USB

USB 输入

NET_DVR_CAM_SDI

SDI 输入

NET_DVR_CAM_HDI

HDI 输入

NET_DVR_CAM_DP

DP 输入，此类型视频综合平台不返回

NET_DVR_CAM_HDTV1

HDTV1 输入

NET_DVR_CAM_JOINT

拼接信号源输入

NET_DVR_CAM_HDBASET

HDBASET 输入

NET_DVR_CAM_DVIT

DVI 双链路，此类型视频综合平台不返回

NET_DVR_CAM_FUSION

多 IPC 拼接

NET_DVR_CAM_VSCREEN

虚拟屏

NET_DVR_CAM_FIBER

光纤

NET_DVR_CAM_3GSDI

3G-SDI

wInputNo

信号源序号，该序号已经扩展，建议使用 *dwInputSignalNo*

sCamName

信号输入源名称

struVideoEffect

视频参数，byColorMode 为 0 时有效

struPuStream

输入流参数，ip 输入时使用

wBoardNum

信号源所在的板卡号，只能获取

wInputIdxOnBoard

信号源在板卡上的位置，只能获取

dwResolution

分辨率

byVideoFormat

视频制式：0- 无，1- NTSC，2- PAL

byStatus

信号源状态：0- 字段无效，1- 有信号，2- 无信号，3- 异常

sGroupName

网络信号源分组的组名

byJointMatrix

关联矩阵：0-不关联，1-关联，当输入信号源为 NET_DVR_CAM_BNC、NET_DVR_CAM_VGA、NET_DVR_CAM_DVI、NET_DVR_CAM_HDMI 中的一种时，该参数有效。

byJointNo

拼接信号源的拼接编号（只能获取）：如果是普通信号源，为 0，表示没有加入拼接；如果是拼接信号源，不能为 0，对应拼接编号

byColorMode

色彩模式：0- 自定义，1- 锐利，2- 普通，3- 柔和，当为自定义时，使用 struVideoEffect 设置

byScreenServer

关联屏幕服务器：0- 不关联，1- 关联

byRes1

保留，置为 0

dwInputSignalNo

输入信号源编号，兼容 wInputNo

byRes

保留，置为 0

Remarks

- 对于本地信号源和矩阵信号源，只有 sCamName 和 struVideoEffect 可以设置；其他参数只能获取，不能设置。
- 设备是否支持 V40 扩展参数 byColorMode 配置能力，可以通过设备能力集进行判断，对应电视墙能力集（[WallAbility](#)），相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：WALL_ABILITY，节点：<InputStreamV40> 中子节点<ColorMode>。

8.69 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

8.70 NET_DVR_IPADDR:IP 地址

```

struct{
    char    sIpV4[16];
    BYTE    sIpV6[128];
}NET_DVR_IPADDR, *LPNET_DVR_IPADDR;

```

Members

sIpV4

设备 IPv4 地址

sIpV6

设备 IPv6 地址

8.71 NET_DVR_IPC_PROTO_LIST_V41:IPC 协议列表

```

struct{
    DWORD    dwSize;
    DWORD    dwProtoNum;
    BYTE     *pBuffer;
    DWORD    dwBufferLen;
    BYTE     byRes[32];
}NET_DVR_IPC_PROTO_LIST_V41, *LPNET_DVR_IPC_PROTO_LIST_V41;

```

Members

dwSize

结构体大小

dwProtoNum

有效的 ipc 协议数目

pBuffer

协议列表缓冲区，保存 dwProtoNum 个 [NET_DVR_PROTO_TYPE](#) 结构内容

dwBufferLen

缓冲区长度

byRes

保留，置为 0

8.72 NET_DVR_JPEGPARA:JPEG 图像信息

```
struct{
    WORD    wPicSize;
    WORD    wPicQuality;
}NET_DVR_JPEGPARA,*LPNET_DVR_JPEGPARA;
```

Members

wPicSize

图片尺寸：0-CIF(352*288/352*240)，1-QCIF(176*144/176*120)，2-4CIF(704*576/704*480)或 D1(720*576/720*486)，3-UXGA(1600*1200)，4-SVGA(800*600)，5-HD720P(1280*720)，6-VGA(640*480)，7-XVGA(1280*960)，8-HD900P(1600*900)，9-HD1080P(1920*1080)，10-2560*1920，11-1600*304，12-2048*1536，13-2448*2048，14-2448*1200，15-2448*800，16-XGA(1024*768)，17-SXGA(1280*1024)，18-WD1(960*576/960*480)，19-1080I(1920*1080)，20-576*576，21-1536*1536，22-1920*1920，0xff-Auto(使用当前码流分辨率)

wPicQuality

图片质量系数：0-最好，1-较好，2-一般

Remarks

参数 wPicQuality 支持的分辨率值可通过 [NET_DVR_GetDeviceAbility](#)（能力集类型：DEVICE_JPEG_CAP_ABILITY）获取：[JPEG 抓图能力集](#)。

8.73 NET_DVR_LED_AREA_COND:LED 区域列表查询条件参数

```
struct{
    DWORD    dwSize;
    DWORD    dwVideoWallNo;
    DWORD    dwLEDAreaNo;
    BYTE     byRes[32];
}NET_DVR_LED_AREA_COND,*LPNET_DVR_LED_AREA_COND;
```

Remarks

dwSize

结构体大小

dwVideoWallNo

电视墙号，从 1 开始

dwLEDAreaNo

LED 区域编号，0 表示查询所有 LED 区域，其他值保留（暂不支持）

byRes

保留，置为 0

8.74 NET_DVR_LED_AREA_INFO:LED 区域信息

```
struct{
    DWORD    dwSize;
```

```

    DWORD                dwLEDAreaNo;
    NET\_DVR\_RECTCFG\_EX   struRect;
    DWORD                dwaOutputNo[MAX_NUM_OUTPUT_CHANNEL];
    BYTE                 byRes[32];
}NET_DVR_LED_AREA_INFO, *LPNET_DVR_LED_AREA_INFO;

```

Members

dwSize

结构体大小

dwLEDAreaNo

LED 区域编号，从 1 开始

struRect

LED 区域矩形坐标

dwaOutputNo

LED 区域包含的输出口列表，每位数组表示一个输出口号(组合)

byRes

保留，置为 0

8.75 [NET_DVR_LED_AREA_INFO_LIST](#):LED 区域信息列表

```

struct{
    DWORD                dwSize;
    DWORD                dwLEDAreaNum;
    LPNET\_DVR\_LED\_AREA\_INFO lpstruBuffer;
    DWORD                dwBufferSize;
    BYTE                 byRes[32];
}NET_DVR_LED_AREA_INFO_LIST, *LPNET_DVR_LED_AREA_INFO_LIST;

```

Members

dwSize

结构体大小

dwLEDAreaNum

LED 区域数量

lpstruBuffer

LED 区域信息列表指针，指向 dwLEDAreaNum 个 [NET_DVR_LED_AREA_INFO](#)

dwBufferSize

缓冲区大小

byRes

保留，置为 0

Remarks

- 获取 LED 区域信息列表时，可以先分配一块足够大的缓冲区 lpstruBuffer，然后调用一次 [NET_DVR_GetSTDConfig](#) 接口，即可将设备发来的数据写入缓冲区；或者先将缓冲区指针 lpstruBuffer 置为 NULL，第一次调用 [NET_DVR_GetSTDConfig](#) 接口时返回 NET_DVR_LED_AREA_INFO 结构的个数 dwLEDAreaNum，然后根据 dwLEDAreaNum 的值申请相应的缓冲区，再次调用 NET_DVR_GetSTDConfig 接口把设备发来的数据写入缓冲区。
- 设备是否支持获取 LED 区域信息列表能力，可以通过设备能力集进行判断，对应电视墙能力集

([WallAbility](#)), 相关接口: [NET_DVR_GetDeviceAbility](#), 能力集类型: WALL_ABILITY, 节点: <isSupportLEDArea>。

8.76 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)。

8.77 NET_DVR_LOCAL_BYTE_ENCODE_CONVERT:字符编码转换参数

```
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。

8.78 NET_DVR_LOCAL_CHECK_DEV:设备在线巡检参数

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。
- 通过 `NET_DVR_RemoteControl`（命令：`NET_DVR_CHECK_USER_STATUS`）可以手动检测设备在线状态。

8.79 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

8.80 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

8.81 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。

8.82 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。

8.83 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。

8.84 NET_DVR_LOG_MATRIX:日志信息结构体

```
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;
    BYTE              byDevSequence;
    BYTE              byMacAddr[MACADDR_LEN];
    BYTE              sSerialNumber[SERIALNO_LEN];
    char              sInfo[LOG_INFO_LEN - MACADDR_LEN - SERIALNO_LEN - 1];
}NET_DVR_LOG_MATRIX,*LPNET_DVR_LOG_MATRIX;
```

Members

strLogTime

日志时间

dwMajorType

日志主类型，定义见[日志主类型列表](#)

dwMinorType

日志次类型，根据不同的主类型的次类型定义见[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
PARA_VIDEOOUT	0x1	视频输出结构配置
PARA_IMAGE	0x2	图像参数结构配置
PARA_ENCODE	0x4	压缩参数结构配置
PARA_NETWORK	0x8	网络参数结构配置
PARA_ALARM	0x10	报警参数结构配置
PARA_EXCEPTION	0x20	异常参数结构配置
PARA_DECODER	0x40	解码器参数结构配置
PARA_RS232	0x80	RS232 参数结构配置
PARA_PREVIEW	0x100	本地预览参数结构配置
PARA_SECURITY	0x200	用户权限参数结构配置
PARA_DATETIME	0x400	本地系统配置
PARA_FRAMETYPE	0x800	帧信息参数结构配置
PARA_CODESPLITTER	0x2000	码分器参数
PARA_RS485	0x2001	RS485 配置信息
PARA_DEVICE	0x2002	设备配置信息
PARA_HARDDISK	0x2003	硬盘配置信息
PARA_AUTOBOOT	0x2004	自动重启配置信息
PARA_HOLIDAY	0x2005	节假日配置信息
PARA_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

byDevSequence

槽位号

byMacAddr

MAC 地址
sSerialNumber
序列号
sInfo
日志附加信息

8.85 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
PARA_VIDEOOUT	0x1	视频输出结构配置
PARA_IMAGE	0x2	图像参数结构配置

PARA_ENCODE	0x4	压缩参数结构配置
PARA_NETWORK	0x8	网络参数结构配置
PARA_ALARM	0x10	报警参数结构配置
PARA_EXCEPTION	0x20	异常参数结构配置
PARA_DECODER	0x40	解码器参数结构配置
PARA_RS232	0x80	RS232 参数结构配置
PARA_PREVIEW	0x100	本地预览参数结构配置
PARA_SECURITY	0x200	用户权限参数结构配置
PARA_DATETIME	0x400	本地系统配置
PARA_FRAMETYPE	0x800	帧信息参数结构配置
PARA_CODESPLITTER	0x2000	码分器参数
PARA_RS485	0x2001	RS485 配置信息
PARA_DEVICE	0x2002	设备配置信息
PARA_HARDDISK	0x2003	硬盘配置信息
PARA_AUTOBOOT	0x2004	自动重启配置信息
PARA_HOLIDAY	0x2005	节假日配置信息
PARA_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

8.86 NET_DVR_MAINBOARD_SERIAL_CFG:主控板串口参数配置

```

struct{
    DWORD    dwSize;
    BYTE     bySerialWorkMode;
    BYTE     byFunType;
    BYTE     byDataBit;
    BYTE     byStopBit;
    DWORD    dwBaudRate;

```

```

BYTE    byParity;
BYTE    byFlowcontrol;
WORD    wProtocol;
BYTE    byVariable;
BYTE    byRes[31];

```

```
}NET_DVR_MAINBOARD_SERIAL_CFG, *LPNET_DVR_MAINBOARD_SERIAL_CFG;
```

Members

dwSize

结构体大小

bySerialWorkMode

串口工作模式：1- RS484，2- RS232

byFunType

串口功能模式：1- 矩阵串口控制，2- 屏幕控制，3- 透明通道模式，4- PTZ 控制，5- 矩阵控制，6- 控制台

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

dwBaudRate

波特：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

byParity

校验：0- 无校验，1- 奇校验，2- 偶校验

byFlowcontrol

流控：0- 无，1- 软流控，2- 硬流控

wProtocol

串口协议，不同串口功能对应串口协议不一样

byVariable

串口是否可变（只获取）：1- 不可变，2-：可变

byRes

保留，置为 0

Remarks

- 根据串口功能不一样，串口协议也不一样：
 - 当串口功能为 1，串口协议为键盘协议，设备支持的键盘协议可以通过 RS232 和 RS485 串口能力集获取，相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：DEVICE_SERIAL_ABILITY，节点：
<KeyBoardProtocol>；
 - 当串口功能为 2，串口协议为屏幕协议；
 - 当串口功能为 3，串口协议字段无效；
 - 当串口功能为 4，串口协议字段无效，且数据位、停止位校验、流控制、波特率等串口参数无效，相关参数通过 NET_DVR_GET_CAMERACHAN_SERIALCFG、NET_DVR_SET_CAMERACHAN_SERIALCFG 获取和配置；
 - 当串口功能为 5，串口协议为矩阵控制协议。
- 上层应先调用获取接口，获取对应的串口是否可变和具体工作模式，再修改设置。对不可变串口，修改工作模式，返回设备串口模式不可配置的错误。

- 使用统一串口编号：综合平台主控的前后面板都有串口，编号统一，不是前后串口各编一个串口号，比如前面板 5 个串口，后面板 5 个串口，那么前面板 1~5，后面板 6~10，不重复。

8.87 NET_DVR_MATRIX_CHAN_INFO_V41:动态解码轮巡通道信息

```
struct{
    BYTE                byEnable;
    BYTE                byStreamMode;
    BYTE                byRes[2];
    NET\_DVR\_DEC\_STREAM\_MODE uDecStreamMode;
}NET_DVR_MATRIX_CHAN_INFO_V41,*LPNET_DVR_MATRIX_CHAN_INFO_V41;
```

Members

byEnable

是否启用：0- 否，1- 启用

byStreamMode

取流模式：0- 无效，1- 通过 IP 或域名取流，2- 通过 URL 取流，3- 通过动态域名解析向设备取流

byRes

保留，置为 0

uDecStreamMode

取流配置信息

8.88 NET_DVR_MATRIX_DEC_CHAN_INFO_V41:当前窗口解码信息

```
struct{
    DWORD                dwSize;
    BYTE                byStreamMode;
    BYTE                byRes1[3];
    NET\_DVR\_DEC\_STREAM\_MODE uDecStreamMode;
    DWORD                dwPlayMode;
    NET\_DVR\_TIME          StartTime;
    NET\_DVR\_TIME          StopTime;
    char                sFileName[128];
    DWORD                dwGetStreamMode;
    NET\_DVR\_MATRIX\_PASSIVEMODE struPassiveMode;
    BYTE                byRes2[32];
}NET_DVR_MATRIX_DEC_CHAN_INFO_V41,*LPNET_DVR_MATRIX_DEC_CHAN_INFO_V41;
```

Members

dwSize

结构体大小

byStreamMode

取流模式：0- 无效，1- 通过 IP 或域名取流，2- 通过 URL 取流，3- 通过动态域名解析向设备取流

byRes1

保留

uDecStreamMode

取流信息

dwPlayMode

解码状态：0-动态解码，1-循环解码，2-按时间回放，3-按文件回放

StartTime

按时间回放开始时间

StopTime

按时间回放结束时间

sFileName

按文件回放文件名

dwGetStreamMode

取流模式：0-未解码，1-主动，2-被动

struPassiveMode

被动解码信息

byRes2

保留

8.89 NET_DVR_MATRIX_DEC_REMOTE_PLAY_EX:远程文件回放解码参数

```
struct{
    DWORD    dwSize;
    DWORD    dwDecChannel;
    BYTE     byAddressType;
    BYTE     byChannelType;
    BYTE     byres[2];
    BYTE     sUserName[NAME_LEN];
    BYTE     sPassword[PASSWD_LEN];
    DWORD    dwChannel;
    BYTE     byStreamId[STREAM_ID_LEN];
    DWORD    dwPlayMode;
    union{
        BYTE     byRes[200];
        NET\_DVR\_IP\_ADDRESS    strulpAddr;
        NET\_DVR\_DDNS\_ADDRESS    struDdnsAddr;
    }unionAddr;
    union{
        BYTE     byRes[128];
        NET\_DVR\_PLAY\_BACK\_BY\_TIME    struPlayBackByTime;
        char     sFileName[128];
    }unionPlayBackInfo;
}NET_DVR_MATRIX_DEC_REMOTE_PLAY_EX, *LPNET_DVR_MATRIX_DEC_REMOTE_PLAY_EX;
```

Members

dwSize
结构体大小

dwDecChannel
子窗口号(组合)

byAddressType
设备地址类型: 0- IP, 1- DDNS 域名

byChannelType
通道类型: 0- 普通通道, 1- 零通道, 2- 流 ID

byres
保留

sUserName
用户名

sPassword
密码

dwChannel
设备通道号

byStreamId
流 ID,此参数在通道类型为流 ID 时有效

dwPlayMode
回放模式: 0- 按文件, 1-按时间

***unionAddr* 为设备地址联合体**

byRes
联合体大小为 200 字节

strulpAddr
设备 IP 地址

struDdnsAddr
设备 DDNS 域名

***unionPlayBackInfo* 为回放参数联合体**

byRes
联合体大小为 128 字节

struPlayBackByTime
按时间回放时文件信息

sFileName
按文件名回放时文件信息

8.90 NET_DVR_MATRIX_LOGO_INFO:LOGO 上传参数

```
struct{
    DWORD    dwSize;
    DWORD    dwLogoSize;
    BYTE     byRes[32];
}NET_DVR_MATRIX_LOGO_INFO,*LPNET_DVR_MATRIX_LOGO_INFO;
```

Members

dwSize

结构体大小

dwLogoSize

LOGO 大小（包括 BMP 的文件头），图片大小限 100K，高和宽须是 32 的倍数

byRes

保留，置为 0

8.91 NET_DVR_MATRIX_DEC_REMOTE_PLAY_STATUS:回放状态信息

```
struct{
    DWORD    dwSize;
    DWORD    dwCurMediaFileLen;
    DWORD    dwCurMediaFilePosition;
    DWORD    dwCurMediaFileDuration;
    DWORD    dwCurPlayTime;
    DWORD    dwCurMediaFileFrames;
    DWORD    dwCurDataType;
    BYTE     res[72];
}NET_DVR_MATRIX_DEC_REMOTE_PLAY_STATUS,*LPNET_DVR_MATRIX_DEC_REMOTE_PLAY_STATUS;
```

Members

dwSize

结构体大小

dwCurMediaFileLen

当前播放的媒体文件长度

dwCurMediaFilePosition

当前播放文件的播放位置

dwCurMediaFileDuration

当前播放文件的总时间

dwCurPlayTime

当前已经播放的时间

dwCurMediaFileFrames

当前播放文件的总帧数

dwCurDataType

当前传输的数据类型：19-文件头，20-流数据，21-播放结束标志

res

保留，置为 0

8.92 NET_DVR_MATRIX_DECCHAN_CONTROL:窗口参数配置

```
struct{
    DWORD    dwSize;
    BYTE     byDecChanScaleStatus;
    BYTE     byDecodeDelay;
    BYTE     byEnableSpartan;
```

```

BYTE    byLowLight;
BYTE    byNoiseReduction;
BYTE    byDefog;
BYTE    byEnableVcaDec;
BYTE    byRes1;
DWORD   dwAllCtrlType;
BYTE    byRes[56];

```

```
}NET_DVR_MATRIX_DECCHAN_CONTROL,*LPNET_DVR_MATRIX_DECCHAN_CONTROL;
```

Members

dwSize

结构体大小

byDecChanScaleStatus

窗口显示缩放控制，1- 缩放显示，0- 真实显示

byDecodeDelay

解码延时：0- 默认，1- 实时性好；2- 实时性较好；3- 实时性中，流畅性中；4- 流畅性较好；5- 流畅性好；0xff- 自动调整

byEnableSpartan

畅显使能：0- 关，1- 开

byLowLight

低照度，0- 关，1~8 代表低照度等级，等级越高强度越大

byNoiseReduction

3D 降噪：0- 关，1- 开，2- 自动

byDefog

透雾，0- 关，1~7 代表透雾等级，等级越高强度越大

byEnableVcaDec

是否启用智能解码：0- 不启用，非 0- 启用

byRes1

保留，置为 0

dwAllCtrlType

所有子窗口一起操作的类型，设置时有效，按位表示，值：0- 不设置，1- 设置
dwAllCtrlType & 0x01：表示开启关闭智能解码

byRes

保留，置为 0

Remarks

- 接口里面 dwDecChan 表示子窗口号(组合)：1 字节墙号 + 1 字节子窗口号 + 2 字节窗口号，将子窗口号置为 0xff，窗口号置为 0xffff，表示对所有已上墙的子窗口进行操作。
- 设备是否支持智能解码能力，可以通过设备能力集进行判断，对应电视墙能力集([WallAbility](#))，相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：WALL_ABILITY，节点：<VcaDecode>。

8.93 NET_DVR_MATRIX_LOGO_CFG:LOGO 参数

```

struct{
    DWORD   dwSize;
    BYTE    byExist;

```

```

    BYTE    byRes1[3];
    DWORD   dwLogoSize;
    BYTE    byLogoName[NAME_LEN];
    BYTE    byRes[32];
}NET_DVR_MATRIX_LOGO_CFG,*LPNET_DVR_MATRIX_LOGO_CFG;

```

Members

dwSize

结构体大小

byExist

LOGO 是否存在（此参数只能获取）：0-不存在，1-存在

byRes1

保留，置为 0

dwLogoSize

LOGO 大小，包括 BMP 的文件头（此参数只能获取）

byLogoName

LOGO 名称

byRes

保留，置为 0

8.94 NET_DVR_MATRIX_LOOP_DECINFO_V41:动态解码轮巡通道参数

```

struct{
    DWORD           dwSize;
    DWORD           dwPoolTime;
    NET\_DVR\_MATRIX\_CHAN\_INFO\_V41 struchanConInfo[MAX_CYCLE_CHAN_V30];
    BYTE            byRes[16];
}NET_DVR_MATRIX_LOOP_DECINFO_V41,*LPNET_DVR_MATRIX_LOOP_DECINFO_V41;

```

Members

dwSize

结构体大小

dwPoolTime

轮巡时间，单位：s

struchanConInfo

动态解码通道信息

byRes

保留，置为 0

8.95 NET_DVR_MATRIX_PASSIVEMODE:被动解码参数

```

struct{
    WORD            wTransProtol;
    WORD            wPassivePort;
    NET\_DVR\_IPADDR struMcastIP;

```

```

    BYTE                byStreamType;
    BYTE                byRes[7];
}NET_DVR_MATRIX_PASSIVEMODE,*LPNET_DVR_MATRIX_PASSIVEMODE;

```

Members

wTransProtol

传输协议：0-TCP，1-UDP，2-MCAST

wPassivePort

UDP 端口，TCP 时端口默认是 8000，不同窗口的 UDP 端口号需设置为不同的值

struMcastIP

TCP/UDP 时无效，MCAST 时为多播地址(暂时不支持多播，保留)

byStreamType

数据播放模式：1- 实时流，2- 文件流

byRes

保留，置为 0

8.96 NET_DVR_MATRIX_STATUS:视频综合平台状态

```

struct{
    DWORD                dwSize;
    BYTE                byMainFrameType;
    BYTE                bySoltNum;
    BYTE                byBoardNum;
    BYTE                byRes;
    NET\_DVR\_MATRIX\_SUBBOARD struMatrixSubboard [MAX_MATRIX_SUBBOARD_NUM];
    BYTE                byRes2[48];
}NET_DVR_MATRIX_STATUS,*LPNET_DVR_MATRIX_STATUS;

```

Members

dwSize

结构体大小

byMainFrameType

机箱类型：1- 18U 机箱(V1.0)，2- 18U 机箱，3- 14U 机箱，4- 11U 机箱，5- 8U 机箱，6- 7U 机箱，7- 5U 机箱

bySoltNum

槽位数

byBoardNum

子板数

byRes

保留，置为 0

struMatrixSubboard

子板状态信息，每个数组对应一块子板，前 byBoardNum 项有效

byRes2

保留，置为 0

8.97 NET_DVR_MATRIX_SUBBOARD: 视频综合平台子板状态信息

```
struct{
    BYTE                byBoardNo;
    BYTE                byPcieConnectStatus;
    BYTE                byRes[2];
    DWORD               dwHardwareVersion;
    DWORD               dwPcieBandwidth;
    DWORD               dwTemperature;
    NET\_DVR\_SUBSYSTEM\_STATUS struSubsystemStatus[MAX_BOARD_SUBSYSTEM_NUM];
    BYTE                byRes2[16];
}NET_DVR_MATRIX_SUBBOARD, *LPNET_DVR_MATRIX_SUBBOARD;
```

Members

byBoardNo

子板号

byPcieConnectStatus

PCI-E 连接状态: 1- 连接, 2- 断开

byRes

保留

dwHardwareVersion

子板硬件版本

dwPcieBandwidth

PCI-E 连接带宽

dwTemperature

子板温度, 单位: 摄氏度

struSubsystemStatus

子系统状态, 每位数组对应一个子系统

byRes2

保留

8.98 NET_DVR_MATRIXDATABASE: 视频综合平台配置文件

```
struct{
    DWORD dwDevType;
    DWORD dwParam;
    BYTE  byFileType;
    BYTE  byRes [3];
} NET_DVR_MATRIXDATABASE, *LPNET_DVR_MATRIXDATABASE;
```

Members

dwDevType

配置文件类型: 1- 域数据库配置文件, 2- 子域数据库配置文件, 3- 平台数据库配置文件, 4- 场景配置文件, 0xff- 最后一次失败的数据库语句

dwParam

参数，代表域号、子域号、平台号，场景配置文件时此值无用

byFileType

文件类型：1- sql 语句，2- db 语句，3- 场景配置文件

byRes

保留，置为 0

8.99 NET_DVR_MATRIXSUBSYSTEMINFO:子系统信息

```
struct{
    BYTE    byMatrixNum;
    BYTE    bySubSystemNum;
    BYTE    byRes [14];
} NET_DVR_MATRIXSUBSYSTEMINFO, *LPNET_DVR_MATRIXSUBSYSTEMINFO;
```

Members

byMatrixNum

视频综合平台号

bySubSystemNum

子系统槽位号

byRes

保留，置为 0

8.100 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

<i>dwSize</i>	结构体大小
<i>struEtherNet</i>	以太网口
<i>struRes1</i>	保留，置为 0
<i>struAlarmHostIpAddr</i>	报警主机 IP 地址
<i>wRes2</i>	保留，置为 0
<i>wAlarmHostIpPort</i>	报警主机端口号
<i>byUseDhcp</i>	是否启用 DHCP：0xff-无效；0-不启用；1-启用
<i>byRes3</i>	保留，置为 0
<i>struDnsServer1IpAddr</i>	域名服务器 1 的 IP 地址
<i>struDnsServer2IpAddr</i>	域名服务器 2 的 IP 地址
<i>byIpResolver</i>	IP 解析服务器域名或 IP 地址（8000 设备不支持域名）
<i>wIpResolverPort</i>	IP 解析服务器端口号
<i>wHttpPortNo</i>	HTTP 端口号
<i>struMulticastIpAddr</i>	多播组地址
<i>struGatewayIpAddr</i>	网关地址
<i>struPPPoE</i>	PPPoE 参数
<i>byRes</i>	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。

8.101 NET_DVR_NETPARAM:网络配置信息

```
struct{
    DWORD                dwSize;
    NET\_DVR\_SINGLE\_NETPARAM struEtherNet[MAX_ETHERNET];
    BYTE                 byRes[64];
}NET_DVR_NETPARAM,*LPNET_DVR_NETPARAM;
```

Members*dwSize*

结构体大小

struEtherNet

以太网口信息

byRes

保留

8.102 NET_DVR_NETSIGNAL_INFO:单个网络信号源信息

```

struct{
    DWORD                dwSize;
    BYTE                 byDevName[NAME_LEN];
    NET\_DVR\_PU\_STREAM\_CFG struPuStream;
    BYTE                 byValid;
    BYTE                 byRes1[3];
    BYTE                 sGroupName[NAME_LEN];
    WORD                 wResolutionX;
    WORD                 wResolutionY;
    BYTE                 byRes2[24];
}NET_DVR_NETSIGNAL_INFO,*LPNET_DVR_NETSIGNAL_INFO;

```

Members*dwSize*

结构体大小

byDevName

网络信号源名称

struPuStream

网络信号源信息，当流媒体结构体 *struStreamMediaSvrCfg* 中的 *byValid* 为 0 时，直接使用 *struDevChanInfo* 中的参数作为网络信号源参数

byValid

网络信号源是否有效：0-无效，删除网络信号源时设为 0；1-有效，添加和修改网络信号源时设为 1

byRes1

保留

sGroupName

网络信号源所在组的组名，网络信号源根据这个组名来分类

wResolutionX

当前分辨率的宽度

wResolutionY

当前分辨率的高度

byRes2

保留

Remarks

- 设置网路信号源时，只能设置“网路信号源名称”，“网路信号源组名”和“传输协议类型”。另外，设置网路信号源只能通过命令 `NET_DVR_SET_NET_SIGNAL` 设置该参数来实现，而不能通过

NET_DVR_SET_INPUTSTREAMCFG_V40 设置。

- 客户端窗口的等比例缩放功能，根据 wResolutionX，wResolutionY 的值来确定所开窗口的宽，高比例。窗口比例有 16:9 和 4:3，具体看分辨率更接近于哪一种窗口比例。

8.103 NET_DVR_NETSIGNALDEV_INFO:网络信号源设备信息

```
struct{
    NET_DVR_NETSIGNAL_DEV_INFO    struDevInfo;
    BYTE                            byRes[816];
}NET_DVR_NETSIGNALDEV_INFO,*LPNET_DVR_NETSIGNALDEV_INFO;
```

Members

struDevInfo

网络信号源信息

byRes

保留

8.104 NET_DVR_NS_RING_CFG:光纤板环网配置

```
struct{
    DWORD                            dwSize;
    BYTE                            byEnable;
    BYTE                            byProtoType;
    BYTE                            byBandWidth;
    BYTE                            byRes1;
    NET_DVR_RING_PORT_PROPERTY    struRingPort[PER_RING_PORT_NUM];
    BYTE                            byRes2[60];
}NET_DVR_NS_RING_CFG, *LPNET_DVR_NS_RING_CFG;
```

Members

dwSize

结构体大小

byEnable

是否启用：0- 不启用，非 0- 启用

byProtoType

协议类型：1- 光端机组网协议(私有)，2- 综合平台组网协议（私有），3- G8032 协议（公有）

byBandWidth

环网带宽类型：1- 百兆，2- 千兆

byRes1

保留，置为 0

struRingPort

环网端口

byRes2

保留，置为 0

Remarks

- 由多台综合平台组成环时，只有一台控制设备，为主设备，其他为从设备。综合平台与光端机组网时，环中只能有一台综合平台设备，综合平台为主设备，光端机为从设备。
- 设备是否支持环网能力或者支持的参数，可以通过设备能力集进行判断，对应环网能力集（NetRingAbility），相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：NET_RING_ABILITY。

8.105 NET_DVR_NS_RING_STATUS:光纤板环网状态

```
struct{
    DWORD    dwSize;
    BYTE     byStatus;
    BYTE     byMasterSlaveProperty ;
    BYTE     byRes[62];
}NET_DVR_NS_RING_STATUS, *LPNET_DVR_NS_RING_STATUS;
```

Members

dwSize

结构体大小

byStatus

环网状态：1- 正常：2- 断开

byMasterSlaveProperty

光纤板主从属性：1- 主，2- 从

byRes

保留

8.106 NET_DVR_OPERATE_DEVICE:设备升级操作参数

```
struct{
    DWORD    dwSize;
    DWORD    dwSubCommand;
    DWORD    dwDeviceIndex;
    BYTE     byRes[44];
}NET_DVR_OPERATE_DEVICE,*LPNET_DVR_OPERATE_DEVICE;
```

Members

dwSize

结构体大小

dwSubCommand

设置设备的命令：1- 设备开始升级，2- 设备停止升级

dwDeviceIndex

设备序号

byRes

保留

8.107 NET_DVR_OPTICAL_CHAN_RELATE_CFG:编码通道关联光口输入源

参数

```
struct{
    DWORD    dwSize;
    BYTE     byEnable;
    BYTE     byRes1[3];
    DWORD    dwDevNo;
    DWORD    dwOpticalPort;
    BYTE     byDevID[MAX_ID_LEN];
    DWORD    dwInputChanNo;
    BYTE     byRes2[64];
}NET_DVR_OPTICAL_CHAN_RELATE_CFG, *LPNET_DVR_OPTICAL_CHAN_RELATE_CFG;
```

Members

dwSize

结构体大小

byEnable

是否关联光端机输入：0- 否，非 0- 是

byRes1

保留，置为 0

dwDevNo

光端机号

dwOpticalPort

光口号

byDevID

光端机物理 ID

dwInputChanNo

光端机输入通道号

byRes2

保留，置为 0

Remarks

- 物理 ID 是节点光端机的唯一标识。当光端机的链路工作模式为环网时，需要配置编码通道关联光端机输入源，根据主口进行关联。
- 目前一块编码板上的前半编码通道只能关联前半的光口，后半的编码通道只能关联后半的光口，不允许交叉关联。
- 设备是否支持光端机接入能力或者支持的参数，可以通过设备能力集进行判断，对应光端机接入能力集（OpticalDevAccessAbility），相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：OPTICAL_DEV_ACCESS_ABILITY。

8.108 NET_DVR_OPTICAL_DEV_CHAN_INFO:光端机设备通道信息

```
struct{
```

```

    DWORD    dwChannel;
    BYTE      byChannelName[NAME_LEN];
    BYTE      bySignal;
    BYTE      bySignalType;
    BYTE      byRes[10];
}NET_DVR_OPTICAL_DEV_CHAN_INFO, *LPNET_DVR_OPTICAL_DEV_CHAN_INFO;

```

Members

dwChannel

通道号，0 表示无效

byChannelName

通道名称

bySignal

是否有信号（只读）：0- 无，1- 有

bySignalType

信号源类型（只读）：1- 标清，2- 高清，有信号时有效

byRes

保留，置为 0

8.109 NET_DVR_OPTICAL_DEV_NODE:节点光端机设备信息

```

struct{
    BYTE      byValid;
    BYTE      byRes1[3];
    DWORD     dwDevNo;
    BYTE      byDevName[NAME_LEN];
    BYTE      byDevID[MAX_ID_LEN];
    NET\_DVR\_OPTICAL\_DEV\_CHAN\_INFO struChannel[MAX_CHANNUM];
    BYTE      byRes2[32];
}NET_DVR_OPTICAL_DEV_NODE, *LPNET_DVR_OPTICAL_DEV_NODE;

```

Members

byValid

是否有效：0-无效，1-有效

byRes1

保留，置为 0

dwDevNo

光端机号

byDevName

节点光端机名称

byDevID

光端机物理 ID，设备唯一标识，只读

struChannel

光端机设备通道信息

byRes2

保留，置为 0

8.110 NET_DVR_OPTICAL_PORT_INFO:光口信息

```
struct {
    DWORD                dwSize;
    BYTE                 byValid;
    BYTE                 byLinkType;
    BYTE                 byPortWorkMode;
    BYTE                 byRes1[1];
    DWORD                dwPairPort;
    NET_DVR_OPTICAL_DEV_NODE struDevInfo[MAX_OPTICAL_DEV_NODE];
    BYTE                 byRes2[32];
}NET_DVR_OPTICAL_PORT_INFO, *LPNET_DVR_OPTICAL_PORT_INFO;
```

Members

dwSize

结构体大小

byValid

是否有信号接入（只读）：0- 无，1- 有

byLinkType

链路类型（只读）：1- 环网，2- 链网

byPortWorkMode

端口工作模式（只读，当链路类型为环网时有效）：0- 无效，1- 主口，2- 辅口

byRes1

保留，置为 0

dwPairPort

配对光口号，链路类型为环网时有效。主口时，此参数为辅口号；辅口时，此参数为主口号

struDevInfo

节点光端机设备信息

byRes2

保留，置为 0

Remarks

- 节点光端机接入编码板实现节点光端机的接入，编码板的每个光口可以串联接入多个节点光端机，每个节点光端机有 4 路信号输入，将节点光端机的某一路信号和编码子系统的编码通道进行绑定，实现对码流的编码压缩。
- 当链路工作模式为环网时，需要占用两个光口来接入信号，一个作为主口，一个作为辅口，正常情况下，使用主口输入的信号，若中途光纤有断开，则会重新组网，变成两条链路。
- 设备是否支持光端机接入能力或者支持的参数，可以通过设备能力集进行判断，对应光端机接入能力集（OpticalDevAccessAbility），相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：OPTICAL_DEV_ACCESS_ABILITY。

8.111 NET_DVR_OSD_INFO:OSD 信息

```
struct{
    BYTE                byEnabled;
```

```

BYTE                byEnabledFlash;
BYTE                byFontSize;
BYTE                byTransparent;
NET\_DVR\_RGB\_COLOR struColor;
WORD                wCoordinateX;
WORD                wCoordinateY;
BYTE                byContent[MAX_LEN_OSD_CONTENT];
BYTE                byRes[32];
}NET_DVR_OSD_INFO, *LPNET_DVR_OSD_INFO;

```

Members

byEnabled

是否使能：0- 不使能，1- 使能

byEnabledFlash

是否闪烁：0- 不闪烁，1- 闪烁

byFontSize

字体大小：1- 大，2- 中，3- 小

byTransparent

透明度，取值范围：0~100

struColor

字体颜色

wCoordinateX

OSD 左上角 X 坐标

wCoordinateY

OSD 左上角 Y 坐标

byContent

OSD 信息

byRes

保留，置为 0

8.112 **NET_DVR_OUT_PARAM:**输出参数

```

struct{
NET\_DVR\_BUF\_INFO struOutBuf;
void*                lpStatusList;
BYTE                byRes[32];
}NET_DVR_OUT_PARAM,LPNET_DVR_OUT_PARAM;

```

Members

struOutBuf

输出参数缓冲区

lpStatusList

状态缓冲区

byRes

保留，置为 0

8.113 NET_DVR_PASSIVEDECODE_CONTROL:被动解码控制参数

```
struct{
    DWORD    dwSize;
    DWORD    dwPlayCmd;
    DWORD    dwCmdParam;
    BYTE     byRes[16];
}NET_DVR_PASSIVEDECODE_CONTROL,*LPNET_DVR_PASSIVEDECODE_CONTROL;
```

Members

dwSize

结构体大小

dwPlayCmd

被动解码控制命令，见下表

宏定义	宏定义值	含义
PASSIVE_DEC_PAUSE	1	被动解码暂停（仅对文件流有效）
PASSIVE_DEC_RESUME	2	恢复被动解码（仅对文件流有效）
PASSIVE_DEC_FAST	3	快速被动解码（仅对文件流有效）
PASSIVE_DEC_SLOW	4	慢速被动解码（仅对文件流有效）
PASSIVE_DEC_NORMAL	5	正常被动解码（仅对文件流有效）
PASSIVE_DEC_ONEBYONE	6	被动解码单帧播放（保留）
PASSIVE_DEC_AUDIO_ON	7	音频开启
PASSIVE_DEC_AUDIO_OFF	8	音频关闭
PASSIVE_DEC_RESETBUFFER	9	清空缓冲区

dwCmdParam

被动解码控制参数

byRes

保留

Remarks

该结构中的第三个参数是否需要输入数值与控制命令有关。

8.114 NET_DVR_PICTURECFG:图片参数

```
struct{
    DWORD    dwSize;
    BYTE     byUseType;
    BYTE     bySequence;
    BYTE     byRes[2];
    NET\_DVR\_BASEMAP\_CFG struBasemapCfg;
    BYTE     sPicName[NAME_LEN];
    DWORD    dwVideoWall;
```

```

    BYTE                byRes2[28];
}NET_DVR_PICTURECFG,*LPNET_DVR_PICTURECFG;

```

Members

dwSize

结构体大小

byUseType

1- 图片当底图使用

bySequence

图片序号

byRes

保留

struBasemapCfg

底图参数，无效（保留）

sPicName

图片名称

dwVideoWall

墙号，1 字节墙号+3 字节保留

byRes2

保留

8.115 NET_DVR_PLAN_CFG:预案配置

```

struct{
    DWORD                dwSize;
    BYTE                byValid;
    BYTE                byWorkMode;
    BYTE                byWallNo;
    BYTE                byPlanNo;
    BYTE                byPlanName[NAME_LEN];
    NET\_DVR\_TIME\_EX      struTime;
    NET\_DVR\_CYCLE\_TIME   struTimeCycle[DAYS_A_WEEK];
    DWORD                dwWorkCount;
    NET\_DVR\_PLAN\_INFO    strPlanEntry[MAX_PLAN_ACTION_NUM];
    BYTE                byRes2[64];
}NET_DVR_PLAN_CFG,*LPNET_DVR_PLAN_CFG;

```

Members

dwSize

结构体大小

byValid

该预案是否有效

byWorkMode

预案工作模式：1- 手动，2- 自动，3- 轮巡

byWallNo

电视墙号，从 1 开始

byPlanNo

预案号，获取预案列表时有效，0 表示无效或不支持

byPlanName

预案名称

struTime

时间，工作模式为自动时有效

struTimeCycle

循环时间，周期为一个星期，年、月、日三个参数不使用。星期取值区间为[0,6]，其中 0 代表星期天，1 代表星期一，以此类推。如：struTimeCycle[0]中的 byValid 的值是 1，表示星期天执行该预案。

dwWorkCount

预案内容执行次数，预案工作模式为手动时该值无效

strPlanEntry

预案执行的内容

byRes2

保留，置为 0

Remarks

- 模式为“手动”时，预案需要手动开启，开启后预案项里面的内容一直循环执行，直到手动停止预案为止。
- 预案工作模式为“自动”时，预案需要手动开启，开启后等到 struTime 中设置的时间点到了以后，预案自动开始执行，当执行 dwWorkCount 次以后自动停止预案。也可以在预案执行过程中手动停止预案。
- 预案工作模式为“轮巡”时，预案需要手动开启，每天检测 struTimeCycle 中设置的时间点，时间到了就开始执行，执行 dwWorkCount 次以后停止预案动作。此时预案仍然在执行，所以无法进行开窗等操作。该工作模式下，预案只能手动停止。

8.116 NET_DVR_PLAN_INFO:预案信息

```
struct{
    BYTE    byValid;
    BYTE    byType;
    WORD    wLayoutNo;
    BYTE    byScreenStyle;
    BYTE    byBaseMapType;
    BYTE    byRes1[2];
    DWORD   dwDelayTime;
    DWORD   dwSerialNo;
    DWORD   dwBaseMapWndNo;
    DWORD   dwBaseMapNo;
    BYTE    byRes2[20];
}NET_DVR_PLAN_INFO,*LPNET_DVR_PLAN_INFO;
```

Members

byValid

该项是否有效

byType

执行类型：

```
enum _NET_DVR_PLAN_OPERATE_TYPE_{
```

```

NET_DVR_SWITCH_LAYOUT = 1,
NET_DVR_SCREEN_POWER_OFF,
NET_DVR_SCREEN_POWER_ON,
NET_DVR_SWITCH_BASEMAP
}NET_DVR_PLAN_OPERATE_TYPE
NET_DVR_SWITCH_LAYOUT
    场景切换(默认)
NET_DVR_SCREEN_POWER_OFF
    关闭大屏幕显示
NET_DVR_SCREEN_POWER_ON
    打开大屏幕显示
NET_DVR_SWITCH_BASEMAP
    底图切换

```

wLayoutNo

场景号， 从 1 开始

byScreenStyle

屏幕型号，开关机所用：1- 低亮，2- 高亮

byBaseMapType

底图类型：1- 图片底图，2- 超高清输入底图，底图切换时有效

byRes1

保留，置为 0

dwDelayTime

一个项的运行时间，单位：秒

dwSerialNo

串口号，屏幕控制时使用

dwBaseMapWndNo

底图窗口号，底图切换时有效

dwBaseMapNo

底图号（底图切换时有效）：底图类型为 1 时，此参数为图片序号；底图类型为 2 时，此参数为超高清输入子系统输入号（1 字节设备号+1 字节子板号+2 字节显示输入序号）

byRes2

保留，置为 0

8.117 NET_DVR_PLAN_LIST:预案列表

```

struct{
    DWORD    dwSize;
    DWORD    dwPlanNums;
    BYTE     *pBuffer;
    BYTE     byWallNo;
    BYTE     byRes1[2];
    DWORD    dwBufLen;
    BYTE     byRes2[64];
}NET_DVR_PLAN_LIST,*LPNET_DVR_PLAN_LIST;

```

Members*dwSize*

结构体大小

dwPlanNums

预案数量

*pBuffer*预案信息，指向 *dwPlanNums* 个 [NET_DVR_PLAN_CFG](#) 结构体大小的缓冲区*byWallNo*

电视墙墙号，从 1 开始

byRes1

保留，置为 0

dwBufLen

所分配缓冲区长度

byRes2

保留，置为 0

Remarks

- 先调用 [NET_DVR_GetPlanList](#)，将该结构体中的 *pBuffer* 置为空且 *dwBufLen* 为 0，此时获取预案个数（*dwPlanNums*）；然后根据接口返回的数量分配 *pBuffer* 的大小，然后再次调用 [NET_DVR_GetPlanList](#)，此时获取预案列表（*pBuffer*）。
- 获取预案列表，设备返回电视墙内所有的预案，包括有效的预案和无效的预案。
- 分布式多屏控制器中，每个电视墙都有自己的预案，所以预案项中的场景也是属于该电视墙内的场景，其他电视墙中的场景不能使用。

8.118 NET_DVR_PLAY_BACK_BY_TIME:回放时间信息

```
struct{
    NET\_DVR\_TIME    StartTime;
    NET\_DVR\_TIME    StopTime;
}NET_DVR_PLAY_BACK_BY_TIME, *LPNET_DVR_PLAY_BACK_BY_TIME;
```

Members*StartTime*

回放开始时间

StopTime

回放结束时间

8.119 NET_DVR_POINT_FRAME:图象区域位置信息

```
struct{
    int    xTop;
    int    yTop;
    int    xBottom;
    int    yBottom;
    int    bCounter;
```

```
}NET_DVR_POINT_FRAME, *LPNET_DVR_POINT_FRAME;
```

Members

xTop

方框起始点的 x 坐标

yTop

方框起始点的 y 坐标

xBottom

方框结束点的 x 坐标

yBottom

方框结束点的 y 坐标

bCounter

画框方向：1- 左上，2- 右上，3- 左下，4- 右下

Remarks

该结构体中的坐标值与当前预览显示框的大小有关，现假设预览显示框为 352*288，我们规定原点为预览显示框左上角的顶点，前四个参数计算方法如下：

$xTop = \text{鼠标当前所选区域的左上角坐标的值} * 255 / 352;$

$xBottom = \text{鼠标当前所选区域的右下角坐标的值} * 255 / 352;$

$yTop = \text{鼠标当前所选区域的左下角坐标的值} * 255 / 288;$

$yBottom = \text{鼠标当前所选区域的右下角坐标的值} * 255 / 288;$

缩小条件：xBottom 减去 xTop 的值大于 2。放大条件：xBottom 减去 xTop 的值大于 0，且 yBottom 减去 yTop 的值大于 0。

8.120 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

8.121 NET_DVR_PROTO_TYPE:协议参数

```
struct{
```

```

DWORD   dwType;
BYTE     byDescribe[DESC_LEN];
}NET_DVR_PROTO_TYPE,*LPNET_DVR_PROTO_TYPE;

```

Members

dwType

协议值

byDescribe

协议描述

Remarks

dwType 取值对应的协议类型可以见 *byDescribe* 协议描述。如果设备支持统一 IPC 协议，对应枚举类型 [NET_DVR_IPC_ENUM_UNIFY](#)；如果不支持统一 IPC 协议，对应枚举类型 [NET_DVR_IPC_ENUM](#)。

设备是否支持统一 IPC 协议，可以通过设备能力集进行判断，对应软硬件能力集（BasicCapability），相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：DEVICE_SOFTWARE_ABILITY，节点：
<SupportUnifiedIPCProtocol>。

```

enum _NET_DVR_IPC_ENUM_{
    ENUM_BUSINESS_INVALID = -1,
    ENUM_BUSINESS_PRIVATE = 0,
    ENUM_BUSINESS_PANASONIC,
    ENUM_BUSINESS_SONY,
    ENUM_BUSINESS_AXIS,
    ENUM_BUSINESS_SANYO,
    ENUM_BUSINESS_BOSCH,
    ENUM_BUSINESS_ZAVIO,
    ENUM_BUSINESS_GRANDEYE,
    ENUM_BUSINESS_PROVIDEO,
    ENUM_BUSINESS_ARECONT,          //9
    ENUM_BUSINESS_ACTI,
    ENUM_BUSINESS_PELCO,
    ENUM_BUSINESS_VIVOTEK,
    ENUM_BUSINESS_INFINOVA,
    ENUM_BUSINESS_DAHUA,           //14
    ENUM_BUSINESS_PRIVT_STD_H264 = 0x20,
    ENUM_BUSINESS_PRIVT_STD_MPEG4,
    ENUM_BUSINESS_SUNELL,          //景阳
    ENUM_BUSINESS_ATEME,
    ENUM_BUSINESS_LAUNCH,          //朗驰
    ENUM_BUSINESS_YAAN,            //雅安
    ENUM_BUSINESS_BLUESKY,         //蓝色星际
    ENUM_BUSINESS_BLUESKYLIMIT,    //蓝色星际
    ENUM_BUSINESS_TDZY,            //天地伟业
    ENUM_BUSINESS_HBGK,            //汉邦高科
    ENUM_BUSINESS_SANTACHI,        //金三立
    ENUM_BUSINESS_HIGHEASY,        //恒忆
    ENUM_BUSINESS_SAMSUNG,

```

```

ENUM_BUSINESS_URL_RTSP = 0x40, //url 类型取流
ENUM_BUSINESS_ONVIF,
ENUM_MAX_BUSINESS_TYPE, //最大厂商类型
}NET_DVR_IPC_ENUM
enum _NET_DVR_IPC_ENUM_UNIFY{
    ENUM_IPC_PROTOCOL_INVALID = -1,
    ENUM_IPC_PROTOCOL_HIKVISION = 0, //海康威视
    ENUM_IPC_PROTOCOL_PANASONIC, //松下
    ENUM_IPC_PROTOCOL_SONY, //索尼

    ENUM_IPC_PROTOCOL_AXIS = 4, //安讯士
    ENUM_IPC_PROTOCOL_SANYO, //三洋
    ENUM_IPC_PROTOCOL_BOSCH, //博世
    ENUM_IPC_PROTOCOL_ZAVIO, //ZAVIO
    ENUM_IPC_PROTOCOL_GRANDEYE, //GRANDEYE
    ENUM_IPC_PROTOCOL_PROVIDEIO, //创视
    ENUM_IPC_PROTOCOL_ARECONT, //爱锐康特
    ENUM_IPC_PROTOCOL_ACTI, //大卫通
    ENUM_IPC_PROTOCOL_PELCO, //帕尔高
    ENUM_IPC_PROTOCOL_VIVOTEK, //晶睿通讯
    ENUM_IPC_PROTOCOL_DAHUA, //大华
    ENUM_IPC_PROTOCOL_SAMSUNG, //三星

    ENUM_IPC_PROTOCOL_PSIA = 17, //PSIA
    ENUM_IPC_PROTOCOL_ONVIF, //ONVIF
    ENUM_IPC_PROTOCOL_BRICKCOM, //金砖通信
    ENUM_IPC_PROTOCOL_GB28181, //GB28181

    ENUM_IPC_PROTOCOL_CANON = 23, //佳能

    ENUM_IPC_PROTOCOL_HUINT = 32, //HUINT
    ENUM_IPC_PROTOCOL_INFINOVA, //英飞拓
    ENUM_IPC_PROTOCOL_HIK_STD_H264, //海康标准 264
    ENUM_IPC_PROTOCOL_HIK_STD_MPEG4, //海康标准 MPEG4
    ENUM_IPC_PROTOCOL_SUNELL, //景阳
    ENUM_IPC_PROTOCOL_ATEME, //安特姆
    ENUM_IPC_PROTOCOL_LAUNCH, //郎驰
    ENUM_IPC_PROTOCOL_YAAN, //亚安
    ENUM_IPC_PROTOCOL_BLUESKY, //蓝色星际
    ENUM_IPC_PROTOCOL_BLUESKYLIMIT, //蓝色星际 LIMIT
    ENUM_IPC_PROTOCOL_TDWY, //天地伟业
    ENUM_IPC_PROTOCOL_HBGK, //汉邦高科
    ENUM_IPC_PROTOCOL_SANTACHI, //金三立
    ENUM_IPC_PROTOCOL_HIGHEASY, //恒亿

```

ENUM_IPC_PROTOCOL_HANBANG ,	//汉邦高科
ENUM_IPC_PROTOCOL_SAMSUNG_3120 ,	//三星 3120
ENUM_IPC_PROTOCOL_SAMSUNG_3080 ,	//三星 3080
ENUM_IPC_PROTOCOL_SAMSUNG_2000,	//三星 2000
ENUM_IPC_PROTOCOL_SAMSUNG_5200 ,	//三星 5200
ENUM_IPC_PROTOCOL_JINGYUAN ,	//警院
ENUM_IPC_PROTOCOL_VIDEOTREC ,	//动力盈科
ENUM_IPC_PROTOCOL_CHENOVA ,	//上海诚丰
ENUM_IPC_PROTOCOL_FENGHUO ,	//烽火
ENUM_IPC_PROTOCOL_ZB_5301 ,	//浙江贝尔 5301
ENUM_IPC_PROTOCOL_ZB_5401 ,	//浙江贝尔 5401
ENUM_IPC_PROTOCOL_HAIXIN,	//海信
ENUM_IPC_PROTOCOL_ZHONGYINGXIN ,	//中瀛鑫
ENUM_IPC_PROTOCOL_AVUN ,	//信诺安达
ENUM_IPC_PROTOCOL_GOVTY ,	//广州吉田
ENUM_IPC_PROTOCOL_SAE ,	//艾里克
ENUM_IPC_PROTOCOL_DONGFANGWANGLI,	//东方网力
ENUM_IPC_PROTOCOL_CHANGHONG,	//长虹
ENUM_IPC_PROTOCOL_H3C ,	//华三
ENUM_IPC_PROTOCOL_BAIAN ,	//百安
ENUM_IPC_PROTOCOL_HAT ,	//华安泰
ENUM_IPC_PROTOCOL_YUANYE,	//远业
ENUM_IPC_PROTOCOL_HIKCARD ,	//海康板卡
ENUM_IPC_PROTOCOL_HAIXINCAP ,	//海信抓拍机
ENUM_IPC_PROTOCOL_WENANCAP ,	//文安抓拍机
ENUM_IPC_PROTOCOL_XUNMEI ,	//讯美
ENUM_IPC_PROTOCOL_BAIWO ,	//佰沃
ENUM_IPC_PROTOCOL_APD ,	//安普达
ENUM_IPC_PROTOCOL_REACHDEV ,	//锐取
ENUM_IPC_PROTOCOL_XUNMEI_DAHUA ,	//讯美大华 OEM
ENUM_IPC_PROTOCOL_HUANGHE,	//黄河设备
ENUM_IPC_PROTOCOL_LIANCHEN ,	//联宸
ENUM_IPC_PROTOCOL_CHENGYE,	//诚业
ENUM_IPC_PROTOCOL_VISIONDIGI ,	//威乾
ENUM_IPC_PROTOCOL_HENGHE ,	//恒河
ENUM_IPC_PROTOCOL_KODAK ,	//柯达
ENUM_IPC_PROTOCOL_AIRONIX,	//奥尼克斯
ENUM_IPC_PROTOCOL_LG,	//乐金
ENUM_IPC_PROTOCOL_HASEE ,	//神州
ENUM_IPC_PROTOCOL_8000ME ,	//8000ME
ENUM_IPC_PROTOCOL_POVITEL ,	//普天博威
ENUM_IPC_PROTOCOL_YIVIEW,	//亿维
ENUM_IPC_PROTOCOL_TIANYUE ,	//天跃
ENUM_IPC_PROTOCOL_HOWELL ,	//浩维

```

ENUM_IPC_PROTOCOL_WAPA,           //波粒
ENUM_IPC_PROTOCOL_SANLE ,         //三乐
ENUM_IPC_PROTOCOL_HIKCARD_ENCRYPTION, //加密海康板卡
ENUM_IPC_PROTOCOL_JUNSDA ,       //宙视达
ENUM_IPC_PROTOCOL_LIYUAN ,       //立元
ENUM_IPC_PROTOCOL_XINCHAN ,      //信产
ENUM_IPC_PROTOCOL_BITE ,         //比特
ENUM_IPC_PROTOCOL_MEIAN,         //美安
ENUM_IPC_PROTOCOL_ROSEEK,        //锐势
ENUM_IPC_PROTOCOL_AEBELL ,       //美电贝尔
ENUM_IPC_PROTOCOL_JSL_ST ,       //金三立 ST
ENUM_IPC_PROTOCOL_VIMICRO,       //中星微电子

ENUM_IPC_PROTOCOL_TYPE,          //最大厂商类型
}NET_DVR_IPC_ENUM_UNIFY, *LPNET_DVR_IPC_ENUM_UNIFY;

```

8.122 NET_DVR_PTZ_3D_SPEED_CONTROL:3D 带速度云台控制参数

```

struct{
    DWORD    dwSize;
    DWORD    dwChannel;
    BYTE     byPSpeed;
    BYTE     byTSpeed;
    BYTE     byZSpeed;
    BYTE     byPDirect;
    BYTE     byTDirect;
    BYTE     byZDirect;
    BYTE     byRes[18];
}NET_DVR_PTZ_3D_SPEED_CONTROL,*LPNET_DVR_PTZ_3D_SPEED_CONTROL;

```

Members

dwSize

结构体大小

dwChannel

子窗口号(组合), 或者编码通道号: (0x80+1 字节子系统号)+1 字节子板号+2 字节编码通道号

byPSpeed

X 轴控制速度, 取值范围: 1~64

byTSpeed

Y 轴控制速度, 取值范围: 1~64

byZSpeed

Z 变倍速度

byPDirect

P 方向: 1- 左, 2- 右

byTDirect

T 方向: 1- 上, 2- 下

byZDirect

Z 方向: 1- ZOOM+, 2- ZOOM-

byRes

保留, 置为 0

8.123 NET_DVR_PTZ_PROTOCOL:云台协议信息

```
struct{
    DWORD      dwType;
    BYTE       byDescribe[DESC_LEN];
}NET_DVR_PTZ_PROTOCOL, *LPNET_DVR_PTZ_PROTOCOL;
```

Members

dwType

协议类型值

byDescribe

协议描述符

8.124 NET_DVR_PTZCFG:云台协议

```
struct{
    DWORD          dwSize;
    NET\_DVR\_PTZ\_PROTOCOL struPtz[PTZ_PROTOCOL_NUM];
    DWORD          dwPtzNum;
    BYTE           byRes[8];
}NET_DVR_PTZCFG, *LPNET_DVR_PTZCFG;
```

Members

dwSize

结构体大小

struPtz

协议信息, 最多 200 种

dwPtzNum

有效的 PTZ 协议数目, 从 0 开始 (即总数为该值加 1)

byRes

保留, 置为

8.125 NET_DVR_PU_STREAM_CFG:动态解码参数

```
struct{
    DWORD          dwSize;
    NET\_DVR\_STREAM\_MEDIA\_SERVER\_CFG struStreamMediaSvrCfg;
    NET\_DVR\_DEV\_CHAN\_INFO          struDevChanInfo;
}NET_DVR_PU_STREAM_CFG, *LPNET_DVR_PU_STREAM_CFG;
```

Members*dwSize*

结构体大小

struStreamMediaSvrCfg

流媒体服务器配置参数

struDevChanInfo

设备通道配置参数

8.126 NET_DVR_PU_STREAM_CFG_V41:动态解码参数

```

struct{
    DWORD                dwSize;
    BYTE                 byStreamMode;
    BYTE                 byRes1[3];
    NET\_DVR\_DEC\_STREAM\_MODE uDecStreamMode;
    BYTE                 byRes2[64];
}NET_DVR_PU_STREAM_CFG_V41,*LPNET_DVR_PU_STREAM_CFG_V41;

```

Members*dwSize*

结构体大小

byStreamMode

取流模式：0- 无效，1- 通过 IP 或域名取流，2- 通过 URL 取流，3- 通过动态域名解析向设备取流

byRes1

保留，置为 0

uDecStreamMode

取流配置信息

byRes2

保留，置为 0

8.127 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

8.128 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

8.129 NET_DVR_RECTCFG:窗口设置

```
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

8.130 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，基准 X 坐标的倍数

dwYCoordinate

矩形左上角 Y 坐标，基准 Y 坐标的倍数

dwWidth

矩形宽度，基准 X 坐标的倍数

dwHeight

矩形高度，基准 Y 坐标的倍数

byRes

保留，置为 0

8.131 NET_DVR_RES_INFO:分辨率信息

```

struct{
    DWORD    dwImageWidth;
    DWORD    dwImageHeight;
}NET_DVR_RES_INFO, *LPNET_DVR_RES_INFO;

```

Members

dwImageWidth

分辨率宽度

dwImageHeight

分辨率高度

8.132 NET_DVR_RGB_COLOR:RGB 颜色信息

```

struct{
    BYTE      byRed;
    BYTE      byGreen;
    BYTE      byBlue;
    BYTE      byRes;
}NET_DVR_RGB_COLOR, *LPNET_DVR_RGB_COLOR;

```

Members

byRed

RGB 颜色三分量中的红色

byGreen

RGB 颜色三分量中的绿色

byBlue

RGB 颜色三分量中的蓝色

byRes

保留，置为 0

8.133 NET_DVR_RING_PORT_PROPERTY:环网端口属性

```
struct{
    BYTE    byPort;
    BYTE    byMasterSlaveProperty;
    BYTE    byPortEthernetType;
    BYTE    byRes[9];
}NET_DVR_RING_PORT_PROPERTY, *LPNET_DVR_RING_PORT_PROPERTY;
```

Members

byPort

端口号

byMasterSlaveProperty

端口主从属性：1- 主，2- 从

byPortEthernetType

端口类型：1- 百兆端口，2- 千兆端口

byRes

保留，置为 0

8.134 NET_DVR_SERIAL_CONTROL:串口信息

```
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

8.135 NET_DVR_SCENE_CONTROL_INFO:场景控制信息

```
struct{
    DWORD dwSize;
    NET\_DVR\_VIDEO\_WALL\_INFO    struVideoWallInfo;
    DWORD                      dwCmd;
    BYTE                       byRes[4];
}NET_DVR_SCENE_CONTROL_INFO, *LPNET_DVR_SCENE_CONTROL_INFO;
```

Members

dwSize

结构体大小

struVideoWallInfo

电视墙信息

dwCmd

场景控制命令，1-场景模式切换（如果要切换的是当前场景，则不进行切换），2-初始化场景（将此场景的配置清空，如果是当前场景，则同时对当前场景进行清屏操作），3-强制切换（无论是否是当前场景，强制切换），4-保存当前模式到某场景，5-删除场景

byRes

保留

Remarks

场景切换保存必须在使能的场景间进行；清空场景后，场景依旧使能，为空场景；场景的使能通过场景模式参数配置接口进行修改；不能删除当前场景。场景切换建议使用 *dwCmd* 为 3 的强制切换。

8.136 NET_DVR_SCHEDTIME:起止时间段参数

```
struct{
    BYTE    byStartHour;
    BYTE    byStartMin;
    BYTE    byStopHour;
    BYTE    byStopMin;
}NET_DVR_SCHEDTIME, *LPNET_DVR_SCHEDTIME;
```

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

8.137 NET_DVR_SCREEN_CONTROL_PARAM: 屏幕控制参数

```
struct{
    NET\_DVR\_INPUT\_INTERFACE\_CTRL    struInputCtrl;
    NET\_DVR\_DISPLAY\_COLOR\_CTRL      struDisplayCtrl;
    NET\_DVR\_DISPLAY\_POSITION\_CTRL   struPositionCtrl;
    BYTE                            byRes[16];
}NET_DVR_SCREEN_CONTROL_PARAM, *LPNET_DVR_SCREEN_CONTROL_PARAM;
```

Members

struInputCtrl

输入接口控制

struDisplayCtrl

显示单元颜色控制

struPositionCtrl

显示单元位置控制

byRes

保留

8.138 NET_DVR_SCREEN_CONTROL_V41: 屏幕控制参数

```
struct{
    DWORD                            dwSize;
    BYTE                             bySerialNo;
    BYTE                             byBeginAddress;
    BYTE                             byEndAddress;
    BYTE                             byProtocol;
    DWORD                            dwCommand;
    NET\_DVR\_SCREEN\_CONTROL\_PARAM     struControlParam;
    BYTE                             byWallNo;
    BYTE                             byRes1[3];
    NET\_DVR\_RECTCFG\_EX               struRect;
    BYTE                             byRes2[28];
}NET_DVR_SCREEN_CONTROL_V41, *LPNET_DVR_SCREEN_CONTROL_V41;
```

Members

dwSize

结构体大小

bySerialNo

串口号

byBeginAddress

左上角屏幕号，从 1 开始

byEndAddress

右下角屏幕号，从 1 开始

byProtocol

串口协议类型: 1-LCD-S1, 2-LCD-S2, 3-LCD-L1, 4-LCD-DLP, 5-LCD-S3, 6-LCD-D20, 7-LCD-L2, 8-LCD-Z1, 9-LCD-D20/D, 10-LCD-D20/S, 11-LCD-D20/P, 12-LCD-D20/T, 13-LCD-D20/F, 14-LCD-D20/DF, 15-LCD-D20/FS, 16-LCD-D20/FP, 17-LCD-D20/FT, 18-LCD-D5022, 19-LCD-D5032, 20-LCD-D5042+, 21-LCD-D20/Z, 0xff-无效(使用预先绑定的协议类型)

dwCommand

控制方法: 1-开, 2-关, 3-屏幕输入源选择, 4-显示单元颜色控制, 5-显示单元位置控制

struControlParam

屏幕控制参数

byWallNo

电视墙号, 从 1 开始

byRes1

保留, 置为 0

struRect

使用坐标范围进行计算, 如同时控制左上角 4 个屏幕, 则坐标参数是(0,0,1920*2,1920*2)

byRes2

保留, 置为 0

Remarks

每次屏幕控制的范围都是一个规则的矩形框。例如, 4*4 的屏幕墙, 如果 byBeginAddress 为 1, byEndAddress 为 5, 则控制器会对第一行第一列, 第二行第二列的两块屏幕发送控制命令, 而屏幕 2、3、4 不会受到控制。各款屏幕支持的功能如下所示:

协议类型	支持的输入口	颜色调整	位置调整	开、关机	备注
LCD-S1	VGA、YPBPR、DVI、BNC、DVI_LOOP、BNC_LOOP, 不支持 SVIDEO、HDMI	亮度、对比度、饱和度、清晰度	不支持	开机、关机 (区域范围要求选择所有屏幕)	必须先选择输入类型再进行颜色调整或位置调整, 否则可能会失败
LCD-S2	VGA、SVIDEO、YPBPR、DVI、BNC, 不支持 DVI_LOOP、BNC_LOOP、HDMI	亮度、对比度、饱和度、清晰度, VGA、YPBPR 和 DVI 不支持饱和度和清晰度调整	水平位置、垂直位置, VGA 和 YPBPR 输入不支持位置调整	开机、关机	LCD-S2 屏幕选择输入类型不同进行颜色调整/位置调整协议也不相同, 必须先选择输入类型再进行颜色调整或位置调整, 否则可能会失败
LCD-L1	VGA、SVIDEO、YPBPR、DVI、BNC、HDMI, 不支持 DVI_LOOP、BNC_LOOP	亮度、对比度、饱和度、清晰度, VGA、DVI 和 HDMI 输入不支持饱和度和清晰度调整	VGA 支持水平位置和垂直位置调整, 其他输入口不支持	开机、关机	必须先选择输入类型再进行颜色调整或位置调整, 否则可能会失败
DLP	VGA、YPBPR、DVI、BNC, 不支持 DVI_LOOP、BNC_LOOP、SVIDEO、HDMI	亮度、对比度	水平位置、垂直位置	开机、关机	必须先选择输入类型再进行颜色调整或位置调整, 否则可能会失败
LCD-S3	/	/	/	/	/
LCD-H1	VGA、YPBPR、DVI、BNC、HDMI	亮度、对比度、饱和度、清晰度, VGA 输入不支持饱和度和清晰度调整	只有 VGA 输入支持位置调整, 其余输入不支持位置调整	开机、关机	必须先选择输入类型再进行颜色调整或位置调整, 否则可能会失败

8.139 NET_DVR_SCREENALARMCFG:多屏控制器报警信息结构体。

```
struct{
    DWORD    dwSize;
    BYTE     byAlarmType;
    BYTE     byBoardType;
    BYTE     bySubException;
    BYTE     byRes1;
    WORD     wStartInputNum;
    WORD     wEndInputNum;
    BYTE     byRes2[16];
}NET_DVR_SCREENALARMCFG, *LPNET_DVR_SCREENALARMCFG;
```

Members

dwSize

结构体大小

byAlarmType

报警类型：1- 子板拔出，2- 子板插入，3- 子系统状态异常，4- 子系统恢复正常，5- 输入源异常，6-温度报警，7- FPGA 版本不匹配，8- 预案开始，9- 预案结束，10- 解码板断网，11- 解码板 IP 地址冲突，12- 风扇异常，13- 解码板断网恢复，14- 解码板 IP 地址冲突恢复

byBoardType

1- 输入板，2- 输出板，3- 主板，4- 背板。报警类型为 1、2、3 时，byBoardType 只能是 1 或 2，报警类型为 6 时，byBoardType 为 3 或 4。

bySubException

输入异常时具体子异常：1- 分辨率改变；2- 接口改变；3- 分辨率错误；4- 分辨率改变导致解码资源不足，关闭该输入源对应窗口；5- 分辨率改变，导致已开窗的缩放比例不在 1/8 到 8 倍范围；6- 分辨率恢复正常；7- 分辨率改变导致输出板数据量超限，设备关闭窗口；8- 分辨率改变导致输入源不支持 9/16 分屏画面，关闭对应子窗口；9- 输出口对应的输入源个数超限

byRes1

保留，置为 0

wStartInputNum

异常输入源（异常起点）

wEndInputNum

异常输入源（异常终点）

byRes2

保留，置为 0

Remarks

发生子异常的时候，由设备关闭窗口，客户端不需要再调用关闭窗口的接口，只需要重新刷新一下窗口列表。

C10H 的开窗缩放比例在 1/8~8 倍的信号源分辨率；C10S 的开窗缩放比例在 1/8~16 倍的信号源分辨率。

8.140 NET_DVR_SCREENZOOM:大屏电子放大参数

```
struct{
```

```

    DWORD          dwSize;
    DWORD          dwScreenNum;
    NET\_DVR\_POINT\_FRAME struPointFrame;
    BYTE           byLayer;
    BYTE           byRes[11];
} NET_DVR_SCREENZOOM, *LPNET_DVR_SCREENZOOM;

```

Members

dwSize
结构体大小

dwScreenNum
子窗口号(组合)

struPointFrame
区域位置信息

byLayer
图层号

byRes
保留

8.141 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

8.142 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

8.143 **NET_DVR_SETUPALARM_PARAM**:报警布防参数

```
struct{
    DWORD    dwSize;
    BYTE     byLevel;
    BYTE     byAlarmInfoType;
    BYTE     byRetAlarmTypeV40;
    BYTE     byRes1[2];
    BYTE     byRetVQDAlarmType;
    BYTE     byRes[11];
}NET_DVR_SETUPALARM_PARAM, *LPNET_DVR_SETUPALARM_PARAM;
```

Members

dwSize

结构体大小

byLevel

布防等级，拼控设备不支持，保留，设为 0

byAlarmInfoType

智能交通报警信息上传类型，拼控设备不支持，保留，设为 0

byRetAlarmTypeV40

拼控设备不支持，保留，设为 0

byRes1

保留，置为 0

byRetVQDAlarmType

VQD 报警上传类型类型，拼控设备不支持，保留，设为 0

byRes

保留，置为 0

8.144 NET_DVR_SHOW_CONTROL_INFO:电视墙显示编号控制

```
struct{
    DWORD    dwSize;
    DWORD    dwDisplayNo;
    BYTE     byEnable;
    BYTE     byChanType;
    BYTE     byRes1[2];
    DWORD    dwWallNo;
    BYTE     byRes2[56];
}NET_DVR_SHOW_CONTROL_INFO,*LPNET_DVR_SHOW_CONTROL_INFO;
```

Members

dwSize

结构体大小

dwDisplayNo

显示输出号(组合), 0xffffffff 表示对所用输出口进行操作

byEnable

使能显示: 0-不显示, 1-显示

byChanType

显示的通道类型: 1-显示输出口号, 2-虚拟屏号

byRes1

保留, 置为 0

dwWallNo

电视墙号(组合): 1 字节电视墙号(高位)+3 字节保留, 显示屏幕序号时有效

byRes2

保留, 置为 0

Remarks

此接口主要应用于接线配置时, 先控制使显示输出号在电视墙上显示, 根据屏幕上显示的显示输出口号配置各个显示输出口的位置坐标, 然后控制显示虚拟屏号(虚拟屏号是设备根据上一步显示输出口的坐标位置计算出来的, 例如一个 3*3 的拼接屏, 虚拟屏号即为 1-9), 如果显示的虚拟屏号从左到右, 从上到下是从 1 开始连续递增的, 那么显示输出口的位置配置是正确的, 如果有错序, 则需要对显示输出口的坐标进行调整。

8.145 NET_DVR_SIGNAL_CUTPARAM:信号源裁剪参数

```
struct{
    DWORD    dwSize;
    DWORD    dwSignalNo;
    DWORD    dwCutTop;
    DWORD    dwCutBottom;
    DWORD    dwCutLeft;
    DWORD    dwCutRight;
    BYTE     byRes2[32];
}
```

```
}NET_DVR_SIGNAL_CUTPARAM, *LPNET_DVR_SIGNAL_CUTPARAM;
```

Members

dwSize

结构体大小

dwSignalNo

信号源序号，通过 [NET_DVR_GetInputSignalList_V40](#) 可以获取设备上所有信号源信息

dwCutTop

上方裁剪像素值，0 代表还原

dwCutBottom

下方裁剪像素值，0 代表还原

dwCutLeft

左边裁剪像素值，0 代表还原

dwCutRight

右边裁剪像素值，0 代表还原

byRes2

保留，置为 0

Remarks

设备是否支持信号源裁剪能力或者支持的参数能力，可以通过设备能力集进行判断，对应电视墙能力集 ([WallAbility](#))，相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：WALL_ABILITY，节点：<InputStreamCut>。

8.146 NET_DVR_SIGNAL_JOINT_CFG:信号源拼接配置

```
struct{
    DWORD    dwSize;
    BYTE     sCamName[NAME_LEN];
    BYTE     byEnable;
    BYTE     byCamMode;
    BYTE     byRows;
    BYTE     byColumns;
    DWORD    dwSignalNo[MAX_SIGNAL_JOINT_NUM];
    DWORD    dwJointNo;
    DWORD    dwSignalNoJoint;
    BYTE     byRes[64];
} NET_DVR_SIGNAL_JOINT_CFG, *LPNET_DVR_SIGNAL_JOINT_CFG;
```

Members

dwSize

结构体大小

sCamName

拼接后信号源名称

byEnable

使能：0- 不使能，0- 使能

byCamMode

拼接后信号源类型，为 NET_DVR_CAM_JOINT

byRows

拼接行数

byColumns

拼接列数

dwSignalNo

信号源序号，前 *byRows** *byColumns* 项有效

dwJointNo

拼接序号（获取时有效）

dwSignalNoJoint

拼接后的信号源号（获取时有效）

byRes

保留，置为 0

Remarks

- 坐标轴是从左上角开始，从左往右，从上往下，*dwSignalNo* 信号源的前 *dwRows** *dwColumns* 项有效，0 代表该位置没有拼接的信号源。
- 获取所有拼接信号源参数时，设备返回所有使能的拼接信号源。对正在使用拼接信号源删除或修改，将返回错误。
- 参与拼接的输入不能再单独使用，一个输入最多只能在一个拼接中。
- 综合平台的 X86 输入和 netra 输入不能进行混合拼接，X86 输入板不能跨板拼接，netra 板可以跨板拼接。C10S 的输入源拼接没有限制，可以不同的输入类型混合拼接。
- 拼接信号源上墙时，使用动态解码接口，作为普通前端源进行上墙，通道类型为拼接通道，通道号使用拼接信号源号，IP 和端口使用主控板地址。
- 各子系统的信号源号按如下规则组合：1 字节设备号+1 字节子板号+2 字节输入编号。
- 该结构体中的使能字段不影响信号源拼接具体参数，而调用 [NET_DVR_RemoteControl](#)(命令：NET_DVR_DEL_SIGNAL_JOINT)删除拼接信号源时，设备将清空拼接参数。
- 设备是否支持信号源拼接能力或者支持的参数能力，可以通过设备能力集进行判断，对应电视墙能力集 ([WallAbility](#))，相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：WALL_ABILITY，节点：<JointSignal>。

8.147 NET_DVR_SINGLE_DEV_INFO:子设备信息

```
struct{
    DWORD                dwSize;
    DWORD                dwSoftVersion;
    DWORD                dwSoftwareBuildDate;
    BYTE                 byUseInSys;
    BYTE                 byDevStatus;
    BYTE                 byDeviceType;
    BYTE                 byRes1[17];
    BYTE                 sDeviceID[DVCS_DEVICEID_LEN];
    BYTE                 sDevName[NAME_LEN];
    NET\_DVR\_SINGLE\_NETPARAM struEtherNet[MAX_ETHERNET];
    BYTE                 sSerialNumber[SERIALNO_LEN];
    NET\_DVR\_DIFFDEV\_INFO struSubDevInfo;
    DWORD                dwDeviceIndex;
    BYTE                 byRes2[48];
}
```



```
}NET_DVR_SINGLE_DEV_INFO,*LPNET_DVR_SINGLE_DEV_INFO;
```

Members

dwSize

结构体大小

dwSoftVersion

设备软件版本号(只能获取): 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

dwSoftwareBuildDate

版本编译日期 Build(只能获取), 0xYYYYMMDD

byUseInSys

是否在系统中使用: 0- 否, 1- 是

byDevStatus

设备当前状态(只能获取): 0- 下线, 1- 上线

byDeviceType

设备类型(只能获取) , 具体定义如下:

```
enum{
    NET_DVR_BNC_PROC   = 1,
    NET_DVR_RGB_PROC   = 2,
    NET_DVR_STP_PROC   = 3,
    NET_DVR_DISP_PROC  = 4,
    NET_DVR_NETSIGNAL  = 5,
    NET_DVR_SDI_PROC   = 6
}NET_DVCS_DEVICE_TYPE
```

NET_DVR_BNC_PROC

BNC 处理器

NET_DVR_RGB_PROC

RGB 处理器

NET_DVR_STP_PROC

转码处理器

NET_DVR_DISP_PROC

显示处理器

NET_DVR_NETSIGNAL

网络信号源

NET_DVR_SDI_PROC

SDI 处理器

byRes1

保留, 置为 0

sDeviceID

设备 ID 标识符

sDevName

设备名称

struEtherNet

设备网络参数

sSerialNumber

设备序列号

struSubDevInfo

设备信息

dwDeviceIndex

设备序号，每个设备唯一确定

byRes2

保留，置为 0

8.148 **NET_DVR_SINGLE_NETPARAM**:单个网络参数

```
struct{
    DWORD          dwSize;
    BYTE           byUseDhcp;
    BYTE           byRes1[3];
    NET\_DVR\_IPADDR struDevIP;
    NET\_DVR\_IPADDR struSubnetMask;
    NET\_DVR\_IPADDR struGateway;
    WORD           wDevPort;
    BYTE           byMACAddr[MACADDR_LEN];
    BYTE           byRes2[16];
}NET_DVR_SINGLE_NETPARAM,*LPNET_DVR_SINGLE_NETPARAM;
```

Members

dwSize

结构体大小

byUseDhcp

是否启用 DHCP: 0xff- 无效, 0- 不启用, 1- 启用

byRes1

保留

struDevIP

设备 IP 地址

struSubnetMask

设备子网掩码

struGateway

设备网关

wDevPort

设备端口号

byMACAddr

MAC 地址（只能获取）

byRes2

保留

8.149 **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

8.150 NET_DVR_SINGLEWALLPARAM:电视墙中屏幕参数

```

struct
{
    DWORD        dwSize;
    BYTE         byEnable;
    BYTE         byRes1[3];
    DWORD        dwWallNum;
    NET\_DVR\_RECTCFG struRectCfg;
    BYTE         byRes2[36];
}NET_DVR_SINGLEWALLPARAM, *LPNET_DVR_SINGLEWALLPARAM;

```

Members

dwSize

结构体大小

byEnable

是否使能: 0- 否, 1- 是

byRes1

保留, 置为 0

dwWallNum

电视墙显示通道号, 用于批量获取。设置时需要和条件参数(lpInBuffer)传入的值相同: 1 字节视频综合平台号+1 字节子板号+1 字节输出口号+1 字节保留(填 0)

struRectCfg

屏幕区域，坐标须为基准坐标的整数倍，宽度和高度值不用设置即为基准值。基准坐标通过能力集 [NET_DVR_GetDeviceAbility](#)（能力集类型：WALL_ABILITY）获取，一般为 1920×1080

byRes2

保留，置为 0

Remarks

屏幕参数获取和设置之前需要先通过接口 [NET_DVR_MatrixGetSubSystemInfo_V40](#) 获取子系统信息（[NET_DVR_SUBSYSTEMINFO_V40](#)），包括子系统所属的子板号（byBelongBoard）、输出口个数（byInterfaceNums）、输出口起始号（byInterfaceStartNum）信息。

8.151 NET_DVR_START_PIC_VIEW_INFO: 图片回显参数

```
struct{
    DWORD    dwSize;
    DWORD    dwSignalIndex;
    DWORD    dwDeviceIndex;
    BYTE     byRes1[12];
    BYTE     byChanIndex;
    BYTE     byRes2[3];
    DWORD    dwScreenNum;
    DWORD    dwLayer;
    DWORD    dwResolution;
    BYTE     byFrame;
    BYTE     bySupportStreamView;
    BYTE     byRes3[14];
}NET_DVR_START_PIC_VIEW_INFO,*LPNET_DVR_START_PIC_VIEW_INFO;
```

Members**dwSize**

结构体大小

dwSignalIndex

信号源序号，当电视墙子窗口回显时，表示子窗口号（组合）。该参数用于多屏控制器，其他设备该值设为 0

dwDeviceIndex

设备序号

byRes1

保留，置为 0

byChanIndex

通道号，从 1 开始

byRes2

保留，置为 0

dwScreenNum

大屏序号，从零开始

dwLayer

图层号，从零开始

dwResolution

回显分辨率,1-QCIF,2-CIF,3-D1

byFrame

帧率, 单位: 帧/秒

bySupportStreamView

客户端是否支持码流回显: 0- 不支持, 1- 支持

byRes3

保留, 置为 0

Remarks

设备支持的回显分辨率、帧率等值通过电视能力集获取 ([NET_DVR_GetDeviceAbility](#), 能力集类型: WALL_ABILITY, 节点: PicViewParam)。

8.152 NET_DVR_STD_CONFIG:配置输入输出参数

```
struct{
    LPVOID    lpCondBuffer;
    DWORD    dwCondSize;
    LPVOID    lpInBuffer;
    DWORD    dwInSize;
    LPVOID    lpOutBuffer;
    DWORD    dwOutSize;
    LPVOID    lpStatusBuffer;
    DWORD    dwStatusSize;
    LPVOID    lpXmlBuffer;
    DWORD    dwXmlSize;
    BYTE      byDataType;
    BYTE      byRes[23];
}NET_DVR_STD_CONFIG,*LPNET_DVR_STD_CONFIG;
```

Members

lpCondBuffer

[in]条件参数(结构体或者 XML 格式), 例如通道号等, 可以为 NULL

dwCondSize

[in]条件参数缓冲区大小

lpInBuffer

[in]输入参数(结构体格式), 获取时为 NULL, 设置时不为 NULL

dwInSize

[in]输入参数缓冲区大小

lpOutBuffer

[out]输出参数(结构体格式), 获取时不为 NULL, 设置时为 NULL

dwOutSize

[in]输出参数缓冲区大小

lpStatusBuffer

[out]返回的状态参数(XML 格式: [ResponseStatus](#)), 获取和设置成功时不会赋值, 如果不需要, 可以置 NULL

dwStatusSize

[in]状态参数缓冲区大小

lpXmlBuffer

[in/out]byDataType = 1 时有效, xml 格式数据内容

dwXmlSize

[in/out]lpXmlBuffer 指向的内存大小: 获取时同时作为输入和输出参数, 获取成功后会修改返回实际长度; 设置时表示实际数据长度【strlen((char*) lpXmlBuffer)】, 而不是整个内存大小

byDataType

[in]输入/输出参数类型: 0- 使用结构体类型 lpInBuffer/lpOutBuffer 有效, 1- 使用 XML 类型 lpXmlBuffer 有效

byRes

[in]保留字节, 置为 0

Remarks

获取配置参数时, lpInBuffer 无效, 设为 NULL。对于不同的配置功能 (NET_DVR_GetSTDConfig 中的配置命令 dwCommand), lpCondBuffer、lpOutBuffer 分别对应不同的内容, 详见 NET_DVR_GetSTDConfig 接口中 "Remarks" 说明。

8.153 NET_DVR_STD_CONTROL:远程控制输入输出参数

```
struct{
    void        *lpCondBuffer;
    DWORD       dwCondSize;
    void        *lpStatusBuffer;
    DWORD       dwStatusSize;
    void        *lpXmlBuffer;
    DWORD       dwXmlSize;
    BYTE        byDataType;
    BYTE        byRes[55];
}NET_DVR_STD_CONTROL,*LPNET_DVR_STD_CONTROL;
```

Members

lpCondBuffer

[in]条件参数(结构体或者 XML 格式),例如通道号等.可以为 NULL

dwCondSize

[in]条件参数缓冲区大小

lpStatusBuffer

[out]返回的状态参数(XML 格式: [ResponseStatus](#)), 控制成功时不会赋值, 如果不需要, 可以置 NULL

dwStatusSize

[in]状态参数缓冲区大小

dwStatusSize

[in]状态参数缓冲区大小

lpXmlBuffer

[in/out]xml 格式数据, byDataType=1 时有效

dwXmlSize

[in]lpXmlBuffer 指向的内存大小: 获取时同时作为输入和输出参数, 获取成功后会修改返回实际长度;

设置时表示实际数据长度【`strlen((char*) lpXmlBuffer)`】，而不是整个内存大小

byDataType

[in]输入/输出参数类型：0- 使用结构体类型 `lpInBuffer/lpOutBuffer` 有效，1- 使用 XML 类型 `lpXmlBuffer` 有效

byRes

[in]保留，置为 0

Remarks

对于不同的控制功能（`NET_DVR_STDControl` 中的控制参数 `dwCommand`），`lpCondBuffer` 对应不同的内容，详见接口 `NET_DVR_STDControl` 中的“Remarks”说明。

8.154 NET_DVR_STREAM_INFO:流 ID 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。全部为 0 时，无效

dwChannel

关联的设备通道。等于 `0xffffffff` 时，如果是设置流的来源信息(`NET_DVR_SET_STREAM_SRC_INFO`)，表示不关联；如果是作为其他如 `NET_DVR_SET_STREAM_RECORD_INFO`、`NET_DVR_SET_STREAM_RECORD_STATUS`、`NET_DVR_SET_MONITOR_VQDCFG` 等配置时的输入条件参数时，表示无效。

byRes

保留，置为 0

Remarks

- 设备不支持流 ID 标识功能，`byID` 值设为 0。
- 对于 X86 码流融合(相机拼接)功能，设备只支持 `NET_DVR_STREAM_INFO` 结构体里的 `dwChannel` 字段，高两字节为融合通道号，低两字节为融合子窗口号。

8.155 NET_DVR_STREAM_MEDIA_SERVER:流媒体服务器参数

```
struct{
    BYTE    byValid;
    BYTE    byRes1[3];
    BYTE    byAddress[MAX_DOMAIN_NAME];
    WORD    wDevPort;
    BYTE    byTransmitType;
```

```

    BYTE    byRes2[5];
}NET_DVR_STREAM_MEDIA_SERVER,*LPNET_DVR_STREAM_MEDIA_SERVER;

```

Members

byValid

是否启用：0- 否，1- 是

byRes1

保留，置为 0

byAddress

IP 或者域名

wDevPort

流媒体服务器端口

byTransmitType

传输协议类型：0- TCP，1- UDP

byRes2

保留，置为 0

8.156 NET_DVR_STREAM_MEDIA_SERVER_CFG:流媒体服务器参数

```

struct{
    BYTE            byValid;
    BYTE            byRes1[3];
    NET\_DVR\_IPADDR  struDevIP;
    WORD            wDevPort;
    BYTE            byTransmitType;
    BYTE            byRes2[69];
}NET_DVR_STREAM_MEDIA_SERVER_CFG,*LPNET_DVR_STREAM_MEDIA_SERVER_CFG;

```

Members

byValid

是否启用流媒体服务器取流：0-不启用，非 0-启用

byRes1

保留，置为 0

struDevIP

流媒体服务器的 IP 地址

wDevPort

流媒体服务器端口

byTransmitType

传输协议类型：0-TCP，1-UDP

byRes2

保留，置为 0

8.157 NET_DVR_STREAM_MEDIA_SERVER_CFG_V41:流媒体服务器参数

```

struct{

```



```

BYTE    byValid;
BYTE    byRes1[3];
BYTE    byAddress[MAX_DOMAIN_NAME];
WORD    wDevPort;
BYTE    byTransmitType;
BYTE    byRes2[69];
}NET_DVR_STREAM_MEDIA_SERVER_CFG_V41,*LPNET_DVR_STREAM_MEDIA_SERVER_CFG_V41;

```

Members

byValid

是否启用流媒体服务器取流：0-不启用，非 0-启用

byRes1

保留，置为 0

byAddress

流媒体服务器 IP 地址或者域名

wDevPort

流媒体服务器端口

byTransmitType

传输协议类型：0-TCP，1-UDP

byRes2

保留，置为 0

8.158 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

8.159 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

8.160 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

8.161 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

8.162 NET_DVR_SUBMATRIXSYSTEMINFO:级联子系统信息

```

struct{
    DWORD    dwSequence;
    BYTE     sAddress[MAX_DOMAIN_NAME];
}

```

```
WORD    wSubMatrixPort;
BYTE    byRes1[6];
BYTE    sUserName[NAME_LEN];
BYTE    sPassword[PASSWD_LEN];
BYTE    byRes2[36];
}NET_DVR_SUBMATRIXSYSTEMINFO,*LPNET_DVR_SUBMATRIXSYSTEMINFO;
```

Members

dwSequence

平台号，0 表示无此平台

sAddress

IP 地址或域名

wSubMatrixPort

子系统端口号

byRes1

保留，置为 0

sUserName

用户名

sPassword

密码

byRes2

保留，置为 0

8.163 NET_DVR_SUBBOARD_INFO:单个子板信息

```
struct{
    DWORD    dwSize;
    BYTE    byBoardType;
    BYTE    byInterfaceNum;
    BYTE    byStatus;
    BYTE    byRes[1];
    DWORD    dwSlotNo;
    BYTE    byRes2[32];
}NET_DVR_SUBBOARD_INFO,*LPNET_DVR_SUBBOARD_INFO;
```

Members

dwSize

结构体大小

byBoardType

子板类型：0- 主板，1- DVI-I 输入，2- YPbPr 输入，3- BNC 输入，4- SDI 输入，5- DVI 双链路输入，6- 普通解码板输入，7- DVI-I 输出，8- SDI 输出，9- 增强型解码板输入，10- DP 输入，11- HDTV 输入，12- HDBaseT 输出，13- HDMI 输入，14- HDBaseT 输入，0xff- 未知类型

byInterfaceNum

输入或者输出接口个数

byStatus

状态：0- 异常，1- 正常

byRes

保留，置为 0

dwSlotNo

所在槽位号(组合): 1 字节子板类型分类 + 3 字节槽位序号,
子板类型分类: 0- 主控, 0x1- 输入类型, 0x2- 输出类型

byRes2

保留，置为 0

Remarks

- 获取子板信息后，对异常的子板，可以调用 [NET_DVR_GetDVRConfig](#)（命令：NET_DVR_GET_SUBBOARD_EXCEPTION）查看子板异常具体信息。

8.164 NET_DVR_SUBSTREAM_SWITCH_CFG:子码流切换配置

struct{

BYTE byAutoSwitchEnable;

BYTE byRes1[3];

WORD wSubWndWidth;

WORD wSubWndHeight;

}NET_DVR_SUBSTREAM_SWITCH_CFG, *LPNET_DVR_SUBSTREAM_SWITCH_CFG;

Members

byAutoSwitchEnable

子码流自动切换: 0- 不启用, 1- 启用

byRes1

保留，置为 0

wSubWndWidth

子窗口宽度设定值，byAutoSwitchEnable 为 1 时有效

wSubWndHeight

子窗口高度设定值，byAutoSwitchEnable 为 1 时有效

Remarks

- 启动子码流自动切换功能后，如果当前子窗口的宽高小于等于设定值（wSubWndWidth、wSubWndHeight）时，设备将自动切换取前端编码设备的子码流进行解码上墙。
- 设备是否支持子码流自动切换能力，可以通过设备能力集进行判断，对应电视墙能力集（[WallAbility](#)），相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：WALL_ABILITY，节点：<SubStreamAutoSwitchCap>。

8.165 NET_DVR_SUBSYSTEM_STATUS:视频综合平台子系统状态信息

struct{

BYTE bySubSystemType;

BYTE bySubSystemNo;

BYTE byOnlineStatus;

BYTE byRes[49];

}NET_DVR_SUBSYSTEM_STATUS, *LPNET_DVR_SUBSYSTEM_STATUS;

Members

bySubSystemType

子系统类型：1- 解码用子系统，2- 编码用子系统，3- 级联输出子系统，4- 级联输入子系统，5- 码分器子系统，6- 报警主机子系统，7- 智能子系统，8- Fpga 解码子系统，9- Fpga 子系统，10- 转码子系统，11- X86 服务器子系统，12- 超高清输入子系统，13- X86 中心服务板，14- X86 业务服务板，15- 多相机拼接子系统，16- 网络光纤交换子系统，17- X86 拼接子系统，18- 普通拼接子系统，0- NULL

bySubSystemNo

子系统号，即所在子板上的编号

byOnlineStatus

网络连接状态，1- 在线，2- 不在线

byRes

保留

8.166 NET_DVR_SUBSYSTEMINFO_V40:视频综合平台子系统参数

```
struct{
    BYTE                bySubSystemType;
    BYTE                byChan;
    BYTE                byLoginType;
    BYTE                bySlotNum ;
    BYTE                byRes1[4]
    NET_DVR_IPADDR      struSubSystemIP;
    NET_DVR_IPADDR      struSubSystemIPMask;
    NET_DVR_IPADDR      struGatewayIpAddr;
    WORD                wSubSystemPort;
    BYTE                byRes2[6];
    BYTE                sUserName[NAME_LEN];
    BYTE                sPassword[PASSWD_LEN];
    char                sDomainName[MAX_DOMAIN_NAME];
    char                sDnsAddresssDomainName[MAX_DOMAIN_NAME];
    BYTE                sSerialNumber[SERIALNO_LEN];
    BYTE                byBelongBoard;
    BYTE                byInterfaceType;
    BYTE                byInterfaceNums;
    BYTE                byInterfaceStartNum;
    BYTE                byDeviceName[20];
    BYTE                byAudioChanNums;
    BYTE                byAudioChanStartNum;
    BYTE                byAudioChanType;
    BYTE                byRes3[33];
}NET_DVR_SUBSYSTEMINFO_V40, *LPNET_DVR_SUBSYSTEMINFO_V40;
```

Members

bySubSystemType

子系统类型(此参数只能获取不能设置)：1-解码子系统，2-编码子系统，3-级联输出子系统，4-级联输入子系统，5-码分器子系统，6-报警主机子系统，7-智能子系统，8-V6 解码子系统，9-V6 子系统，

10-转码子系统, 11-X86 服务器子系统, 12-超高清输入子系统, 13-X86 中心服务板, 14-X86 业务服务板, 15-网络光纤交换子系统, 16-多相机拼接子系统, 17-X86 拼接子系统, 18-普通拼接子系统, 0-NULL

byChan

子系统通道数, 对于码分子系统则代表 RS485 串口数量, 此参数只能获取不能设置

byLoginType

注册类型, 1-直连, 2-DNS, 3-花生壳

bySlotNum

槽位号, 此参数只能获取

byRes1

保留

struSubSystemIP

子系统 IP 地址

struSubSystemIPMask

子系统子网掩码

struGatewayIpAddr

子系统网关地址

wSubSystemPort

子系统端口号

byRes2

保留

sUserName

子系统用户名, 此参数只能获取不能设置

sPassword

子系统密码, 此参数可修改

sDomainName

子系统域名, 此参数可修改

sDnsAddresssDomainName

子系统 DNS, 可以是域名或者 IP 地址

sSerialNumber

子系统序列号, 此参数只能获取不能设置

byBelongBoard

所属子板号, 从 1 开始, 0xff 表示无效

byInterfaceType

接口类型: 1-BNC, 2-VGA, 3-HDMI, 4-DVI, 5-SDI, 6-光纤(FIBER), 7-RGB, 8-YprPb, 9-VGA/HDMI/DVI 自适应, 10-3GSDI, 11-VGA/DVI 自适应, 13-HDBaseT, 14-DP, 15-DVIT, 0xff 表示无效(超高清输入或虚拟屏等无实际物理接口的类型)

byInterfaceNums

接口个数, 0xff 表示无效

byInterfaceStartNum

接口起始号, 0xff 表示无效

byDeviceName

子系统名称

byAudioChanNums

音频通道个数

byAudioChanStartNum

音频通道起始号

byAudioChanType

音频通道类型：0-无效，1-音频输入，2-音频输出

byRes3

保留

8.167 **NET_DVR_SUBWND_DECODE_OSD**:子窗口解码 OSD 信息

```
struct{
    DWORD          dwSize;
    DWORD          dwSubWndNo;
    DWORD          dwOSDNums;
    NET\_DVR\_OSD\_INFO struOSDList[MAX_NUM_OSD_ONE_SUBWND];
    BYTE           byRes[32];
}NET_DVR_SUBWND_DECODE_OSD, *LPNET_DVR_SUBWND_DECODE_OSD;
```

Members

dwSize

结构体大小

dwSubWndNo

子窗口号，从 1 开始

dwOSDNums

该子窗口配置的 OSD 信息的个数

struOSDList

OSD 信息列表，每位数组表示一个 OSD 信息，前 dwOSDNums 位数组有效

byRes

保留，置为 0

8.168 **NET_DVR_SUBWND_DECODE_OSD_LIST**:子窗口解码 OSD 信息列表

```
struct
{
    DWORD          dwSize;
    NET\_DVR\_SUBWND\_DECODE\_OSD struSubWndList[MAX_NUM_SPLIT_WND];
    BYTE           byRes[32];
}NET_DVR_SUBWND_DECODE_OSD_LIST, *LPNET_DVR_SUBWND_DECODE_OSD_LIST;
```

Members

dwSize

结构体大小

struSubWndList

所有子窗口的解码 OSD 信息，每位数组表示一个子窗口

byRes

保留，置为 0

8.169 NET_DVR_TEMPERATUREALARMCFG:温度超限报警配置参数

```
struct{
    BYTE                byEnableTemperatureAlarm;
    BYTE                byRes1[3];
    int                 iTemperatureUpLimited;
    int                 iTemperatureDownLimited;
    NET\_DVR\_HANDLEEXCEPTION\_V30 struTempHandleType;
    NET\_DVR\_SCHEDTIME      struAlarmTime[MAX_DAYS][MAX_TIMESEGMENT_V30];
    BYTE                byRes2[32];
}NET_DVR_TEMPERATUREALARMCFG, *LPNET_DVR_TEMPERATUREALARMCFG;
```

Members

byEnableTemperatureAlarm

温度超限报警使能：0-报警，1-不报警

byRes1

保留

iTemperatureUpLimited

温度上限值，精确到小数点后两位，如 1234 表示温度 12.34 度

iTemperatureDownLimited

温度下限值，精确到小数点后两位，如 1234 表示温度 12.34 度

struTempHandleType

处理方式

struAlarmTime

布防时间

byRes2

保留

8.170 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

8.171 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

8.172 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

8.173 NET_DVR_UNITEDMATRIXSYSTEMINFO:级联配置信息

```

struct
{
    DWORD                dwSize;
    NET\_DVR\_SUBMATRIXSYSTEMINFO struMatrixInfo[MAX_UNITEDMATRIX_NUM];
    BYTE                 byRes[32];
}NET_DVR_UNITEDMATRIXSYSTEMINFO, *LPNET_DVR_UNITEDMATRIXSYSTEMINFO;

```

Members

dwSize

结构体大小

struMatrixInfo

视频综合平台信息

byRes

保留

8.174 NET_DVR_UPGRADE_STATE:设备升级状态信息

```

struct{
    DWORD    dwProgress;
    BYTE     byState;
    BYTE     byRes2[31];
}NET_DVR_UPGRADE_STATE, *LPNET_DVR_UPGRADE_STATE;

```

Members

dwProgress

升级进度：0~100

byState

升级阶段：1-升级完成，2-正在升级，3-升级失败

byRes2

保留

8.175 NET_DVR_USER_INFO_V40:单用户参数

```
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

8.176 **NET_DVR_USER_LOGIN_INFO**: 用户登录参数

```
struct{
    char                sDeviceAddress[NET_DVR_DEV_ADDRESS_MAX_LEN];
    BYTE                byRes1;
    WORD                wPort;
    char                sUserName[NET_DVR_LOGIN_USERNAME_MAX_LEN];
    char                sPassword[NET_DVR_LOGIN_PASSWD_MAX_LEN];
    fLoginResultCallBack cbLoginResult;
    void                *pUser;
    BOOL                bUseAsynLogin;
    BYTE                byRes2[128];
}NET_DVR_USER_LOGIN_INFO,*LPNET_DVR_USER_LOGIN_INFO;
```

Members

sDeviceAddress

设备地址，IP 或者普通域名

byRes1

保留，设为 0

wPort

设备端口号，例如：8000

sUserName

登录用户名，例如：admin

sPassword

登录密码，例如：12345

cbLoginResult

登录状态回调函数，bUseAsynLogin 为 1 时有效

pUser

用户数据

bUseAsynLogin

是否异步登录：0- 否，1- 是

byRes2

保留，置为 0

Callback Function

```
typedef void(CALLBACK *fLoginResultCallBack)(LONG IUserID, DWORD dwResult,
LPNET\_DVR\_DEVICEINFO\_V30 lpDeviceInfo, void *pUser);
```

Callback Function Parameters

IUserID

[out] 用户 ID, NET_DVR_Login_V40 的返回值

dwResult

[out] 登录状态: 0- 异步登录失败, 1- 异步登录成功

lpDeviceInfo

[out] 设备信息, 设备序列号、通道、能力等参数

pUser

[out] 用户数据

8.177 NET_DVR_USER_V40:用户参数

```
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

8.178 NET_DVR_USING_SERIALPORT:当前使用串口号信息

```
struct
{
    DWORD    dwSize ;
    DWORD    dwSerialPort;
    BYTE     byProtocolType;
    BYTE     byRes2[31];
}NET_DVR_USING_SERIALPORT,*LPNET_DVR_USING_SERIALPORT;
```

Members

dwSize

结构体大小

dwSerialPort

当前使用串口号

byProtocolType

串口协议类型: 1-LCD-S1, 2-LCD-S2, 3-LCD-L1, 4-LCD-DLP, 5-LCD-S3, 6-LCD-D20, 7-LCD-L2, 8-LCD-Z1, 9-LCD-D20/D, 10-LCD-D20/S, 11-LCD-D20/P, 12-LCD-D20/T, 13-LCD-D20/F, 14-LCD-D20/DF, 15-LCD-D20/FS, 16-LCD-D20/FP, 17-LCD-D20/FT, 18-LCD-D5022, 19-LCD-D5032, 20-LCD-D5042+

byRes2

保留, 置为 0

8.179 NET_DVR_VCS_USER_INFO:用户信息

```

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];
    NET\_DVR\_RECTCFG\_EX    struManageRegion;
    BYTE                byWallNo;
    BYTE                byRes2[107];
}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-虚拟 LED 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-导入导出皮配置，bit13-日志查询，bit14-远程重启，bit15-屏幕区域管理，bit16-信号源权限

struInputParam

关联的输入设备

struOutputParam

关联的输出显示屏

struManageRegion

屏幕管理区域

byWallNo

开启屏幕管理区域的墙号（对于只支持单墙的设备，此字段无效）

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。
- 设置操作员用户参数时，用户名（**sUserName**）为空表示删除相应用户。
- 管理员具有对所有设备操作的权限，能够给普通用户分配输入设备，每个输入设备都只能被一个普通用户关联。
- 管理员能够给用户划分不同操作屏幕区域（关联输出显示屏），普通用户只能在该区域内进行窗口操作。普通用户只有开窗口和移动窗口操作权限，没有其他权限。
- 信号源权限，只对本地管理信号源有效，网络信号源统一由客户端管理。
- 限定屏幕区域开启时，只影响用户开窗、漫游权限，其他权限不受限定屏幕区域影响。
- 如果开启电视墙区域限制，此用户只能在指定电视墙的指定区域内进行窗口相关操作。

8.180 NET_DVR_VEDIOPLATLOG:视频综合平台日志搜索参数

```
struct{
    BYTE    bySearchCondition;
    BYTE    byDevSequence;
    BYTE    sSerialNumber[SERIALNO_LEN];
    BYTE    byMacAddr[MACADDR_LEN];
}NET_DVR_VEDIOPLATLOG, *LPNET_DVR_VEDIOPLATLOG;
```

Members

bySearchCondition

搜索条件，0-按槽位号搜索，1-按序列号搜索，2-按 MAC 地址进行搜索

byDevSequence

槽位号：0~79：对应子系统的槽位号；0xff：表示搜索所有的日志，包括中央主控板和所有编解码子系统；0xfe：表示搜索中央主控板上的日志；0xfd：表示搜索子域日志；0xfc：表示搜索域日志

sSerialNumber

序列号

byMacAddr

MAC 地址

8.181 NET_DVR_VIDEO_WALL_INFO:电视墙信息

```
struct{
    DWORD    dwSize;
    DWORD    dwWindowNo;
    DWORD    dwSceneNo;
```



```

    BYTE        byRes[20];
}NET_DVR_VIDEO_WALL_INFO,*LPNET_DVR_VIDEO_WALL_INFO;

```

Members

dwSize

结构体大小

dwWindowNo

窗口号(组合): 1 字节墙号+1 字节保留+2 字节窗口号

dwSceneNo

场景号

byRes

保留, 置为 0

Remarks

场景配置文件的导入导出只使用 *dwWindowNo* 的最高字节 (墙号), 导入导出的配置文件为电视墙当前场景。

8.182 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]

8.183 NET_DVR_VIDEOPLATFORMALARMCFG:视频综合平台报警参数

```
struct{
    DWORD                dwSize;
    NET\_DVR\_TEMPERATUREALARMCFG    struTempAlarmCfg;
    NET\_DVR\_BOARDALARMCFG        struBoardAlarmCfg;
    NET\_DVR\_HANDLEEXCEPTION\_V30    struHandleException[MAX_EXCEPTIONNUM];
    BYTE                byRes[32];
}NET_DVR_VIDEOPLATFORMALARMCFG, *LPNET_DVR_VIDEOPLATFORMALARMCFG;
```

Members

dwSize

结构体大小

struTempAlarmCfg

温度超限报警参数结构体

struBoardAlarmCfg

热插拔报警参数结构体

struHandleException

其他异常报警处理: 数组 0- 风扇异常, 数组 1- 子系统异常, 数组 2- IP 地址冲突, 数组 3- 网线断, 数组 4- 非法访问, 数组 5- 光纤异常, 数组 6- 前面板异常

byRes

保留

8.184 NET_DVR_VIDEOPLATFORM_ALRAMINFO:视频综合平台报警信息

```
struct{
    DWORD                dwSize;
    BYTE                byAlarmType;
    BYTE                byBoardNum;
    BYTE                byRes1[2];
    BYTE                bySubSystemChan[8];
    int                 iTemperature;
    BYTE                byMainboardSeq;
    BYTE                byRes2[3];
    BYTE                byFanSequence[32];
    NET\_DVR\_MATRIXSUBSYSTEMINFO    struInputNote;
    NET\_DVR\_MATRIXSUBSYSTEMINFO    struOutputNote;
    BYTE                byRes3[68];
}NET_DVR_VIDEOPLATFORM_ALRAMINFO, *LPNET_DVR_VIDEOPLATFORM_ALRAMINFO;
```

Members

dwSize

结构体大小

byAlarmType

报警类型：0- 拔出子板，1- 插入子板，2- 温度超上限报警，3- 温度超下限报警，4- 风扇状态异常，5- 子系统异常，6- 非法访问，7- 光纤异常报警，8- 前面板异常报警

byBoardNum

子板号，从 1 起，B10 不支持该字段，无效，置为 0xff

byRes1

保留，置为 0

bySubSystemChan

子板槽位号，一个子板中有 8 个子系统，其分别对应的槽位号，子系统异常时，数组下标对应子系统序号，例如：**bySubSystemChan[0]**表示子系统 1 的子板槽位号。

iTemperature

温度超限报警时有用，温度值精确到小数点后两位，如 1234 表示温度 12.34 度

byMainboardSeq

温度超限报警时有用，0-非主板，1-主板

byRes2

保留，置为 0

byFanSequence

风扇异常时风扇号

struInputNote

光纤异常报警时使用，输入结点

struOutputNote

光纤异常报警使用，输出结点

byRes3

保留，置为 0

8.185 **NET_DVR_VIDEOWALLDISPLAYMODE**:电视墙拼接模式配置

struct

```
{
    DWORD          dwSize;
    BYTE           byEnable;
    BYTE           byRes1[3];
    NET\_DVR\_RECTCFG\_EX struRect;
    BYTE           sName[NAME_LEN];
    BYTE           byRes2[100];
}NET_DVR_VIDEOWALLDISPLAYMODE, *LPNET_DVR_VIDEOWALLDISPLAYMODE;
```

Members

dwSize

结构体大小

byEnable

使能：0- 禁用，1- 启用

byRes1

保留，置为 0

struRect

坐标范围，依据基准坐标计算出支持的拼控模式

sName

电视墙名称

byRes2

保留，置为 0

8.186 NET_DVR_VIDEOWALLDISPLAYPOSITION:电视墙输出位置配置

```
struct{
    DWORD                dwSize;
    BYTE                 byEnable;
    BYTE                 byRes1[3];
    DWORD                dwVideoWallNo;
    DWORD                dwDisplayNo;
    NET\_DVR\_RECTCFG\_EX   struRectCfg;
    BYTE                 byRes2[64];
}NET_DVR_VIDEOWALLDISPLAYPOSITION,*LPNET_DVR_VIDEOWALLDISPLAYPOSITION;
```

Members

dwSize

结构体大小

byEnable

使能：0- 禁用，1- 启用

byRes1

保留，置为 0

dwVideoWallNo

电视墙号(组合)

dwDisplayNo

显示输出号(组合)，批量获取全部时有效

struRectCfg

位置坐标，须为基准坐标（通过能力集获取）的整数倍，宽度和高度值不用设置，即为基准值

byRes2

保留，置为 0

8.187 NET_DVR_VIDEOWALLWINDOWPOSITION:电视墙窗口信息

```
struct{
    DWORD                dwSize;
    BYTE                 byEnable;
    BYTE                 byWndOperateMode;
    BYTE                 byRes1[6];
    DWORD                dwWindowNo;
    DWORD                dwLayerIndex;
    NET\_DVR\_RECTCFG\_EX   struRect;
```

```

NET_DVR_RECTCFG_EX    struResolution;
DWORD                 dwXCoordinate;
DWORD                 dwYCoordinate;
BYTE                  byRes2[36];

```

```

}NET_DVR_VIDEOWALLWINDOWPOSITION, *LPNET_DVR_VIDEOWALLWINDOWPOSITION;

```

Members

dwSize

结构体大小

byEnable

窗口使能：0- 禁用，1- 启用

byWndOperateMode

窗口操作模式：0- 统一坐标，1- 分辨率坐标

byRes1

保留，置为 0

dwWindowNo

窗口号(组合)，批量获取全部时有效

dwLayerIndex

窗口相对应的图层号，只能获取，不能设置，图层号由设备关联

struRect

目的窗口坐标区域(相对显示墙)，byWndOperateMode 为 0 时有效，使用统一坐标

struResolution

目的窗口分辨率坐标，byWndOperateMode 为 1 时有效，使用分辨率坐标

dwXCoordinate

LED 区域左上角 X 坐标（统一坐标，相对于电视墙的位置），byWndOperateMode 为 1 时有效

dwYCoordinate

LED 区域左上角 Y 坐标（统一坐标，相对于电视墙的位置），byWndOperateMode 为 1 时有效

byRes2

保留，置为 0

Remarks

- 电视墙中开窗口默认置顶，窗口置顶置底操作通过 [NET_DVR_RemoteControl](#) 实现。
- 关于物理分辨率开窗：
 - 设备是否支持分辨率持物理分辨率开窗模式，可以通过设备能力集进行判断，对应电视墙能力集 ([WallAbility](#))，相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：WALL_ABILITY，节点：[<isSupportPhysicsResolutionWnd>](#)。
 - 如果设备支持物理分辨率开窗，通过电视墙能力集中[<VoutResource>](#)的子节点[<LEDResolution>](#)判断各个输出口是否支持设置 LED 分辨率。支持的情况下，可以通过 [NET_DVR_GetDeviceConfig](#)(命令：[NET_DVR_WALLOUTPUT_GET](#))、[NET_DVR_SetDeviceConfig](#)(命令：[NET_DVR_WALLOUTPUT_SET](#))获取和设置显示输出参数([NET_DVR_WALLOUTPUTPARAM](#))里面的 LED 分辨率。设置了 LED 分辨率的显示输出口即属于 LED 区域。
 - 根据电视墙能力集里面的[<isSupportLEDArea>](#)的节点可以判断设备是否支持获取 LED 区域列表，若支持则可以调用 [NET_DVR_GetSTDConfig](#)(命令：[NET_DVR_GET_LED_AREA_INFO_LIST](#))获取 LED 区域坐标信息，再根据 LED 区域对应的起点坐标进行按物理分辨率开窗。若设备不支持获取 LED 区域列表，则默认整个电视墙都是 LED 区域，即 LED 区域左上角 X 坐标、LED 区域左上角 Y 坐标均为 0。

8.188 NET_DVR_VIRTUAL_SCREEN_CFG:超高清输入子系统配置

```
struct{
    DWORD   dwSize;
    DWORD   dwResolution;
    BYTE     byRes2[64];
}NET_DVR_VIRTUAL_SCREEN_CFG, *LPNET_DVR_VIRTUAL_SCREEN_CFG;
```

Members

dwSize

结构体大小

dwResolution

分辨率，MAKE_RESOLUTION 宏定义的值，从电视墙能力获取支持的分辨率值

宏定义	宏定义值	含义
NOT_AVALIABLE	MAKE_RESOLUTION(0,0,0,0)	无
SVGA_60HZ	MAKE_RESOLUTION(0, 800, 600, 60)	SVGA:800*600, 频率: 60HZ
SVGA_75HZ	MAKE_RESOLUTION(0, 800, 600, 75)	SVGA:800*600, 频率: 75HZ
XGA_60HZ	MAKE_RESOLUTION(0, 1024, 768, 60)	XGA:1024*768, 频率: 60HZ
XGA_75HZ	MAKE_RESOLUTION(0, 1024, 768, 75)	XGA:1024*768, 频率: 75HZ
SXGA_60HZ	MAKE_RESOLUTION(0, 1280, 1024, 60)	SXGA:1280*1024, 频率: 60HZ
SXGA2_60HZ	MAKE_RESOLUTION(0, 1280, 960, 60)	SXGA2:1280*960, 频率: 60HZ
_720P_60HZ	MAKE_RESOLUTION(0, 1280, 720, 60)	720P:1280*720, 频率: 60HZ
_720P_50HZ	MAKE_RESOLUTION(0, 1280, 720, 50)	720P:1280*720, 频率: 50HZ
_1080I_60HZ	MAKE_RESOLUTION(1, 1920, 1080, 60)	1080I:1920*1080, 频率: 60HZ
_1080I_50HZ	MAKE_RESOLUTION(1, 1920, 1080, 50)	1080I:1920*1080, 频率: 50HZ
_1080P_60HZ	MAKE_RESOLUTION(0, 1920, 1080, 60)	1080P:1920*1080, 频率: 60HZ
_1080P_50HZ	MAKE_RESOLUTION(0, 1920, 1080, 50)	1080P:1920*1080, 频率: 50HZ
_1080P_30HZ	MAKE_RESOLUTION(0, 1920, 1080, 30)	1080P:1920*1080, 频率: 30HZ
_1080P_25HZ	MAKE_RESOLUTION(0, 1920, 1080, 25)	1080P:1920*1080, 频率: 25HZ
_1080P_24HZ	MAKE_RESOLUTION(0, 1920, 1080, 24)	1080P:1920*1080, 频率: 24HZ
UXGA_60HZ	MAKE_RESOLUTION(0, 1600, 1200, 60)	UXGA:1600*1200, 频率: 60HZ
UXGA_30HZ	MAKE_RESOLUTION(0, 1600, 1200, 30)	UXGA:1600*1200, 频率: 30HZ
WSXGA_60HZ	MAKE_RESOLUTION(0, 1680, 1050, 60)	WSXGA:1680*1050, 频率: 60HZ
WUXGA_60HZ	MAKE_RESOLUTION(0, 1920, 1200, 60)	WUXGA:1920*1200, 频率: 60HZ
WUXGA_30HZ	MAKE_RESOLUTION(0, 1920, 1200, 30)	WUXGA:1920*1200, 频率: 30HZ
WXGA_60HZ	MAKE_RESOLUTION(0, 1360, 768, 60)	WXGA:1360*768, 频率: 60HZ
SXGA_PLUS_60HZ	MAKE_RESOLUTION(0, 1400, 1050, 60)	SXGA_PLUS:1400*1050, 频率: 60HZ

byRes2

保留，置为 0

Remarks

```
#define MAKE_RESOLUTION(_interlace_, _width_, _height_, _fps_) \
((( _interlace_ )<<28) | ((( _width_ )>>3)&0x1ff)<<19) | ((( _height_ )>>1)&0x7ff)<<8) | (( _fps_ )&0xff))
#define GET_RES_INTERLACE(_res_) ((( _res_ )>>28)&0x1)
#define GET_RES_WIDTH(_res_) ((( _res_ )>>19)&0x1ff)<<3)
#define GET_RES_HEIGHT(_res_) ((( _res_ )>>8)&0x7ff)<<1)
#define GET_RES_FPS(_res_) (( _res_ )&0xff)
```

8.189 NET_DVR_VIRTUALLED_PARAM: 虚拟 LED 参数

```
struct{
    DWORD                dwSize;
    BYTE                 byEnable;
    BYTE                 byDispMode;
    BYTE                 byRes1[6];
    NET_DVR_RGB_COLOR    struContentColor;
    NET_DVR_RGB_COLOR    struBackColor;
    NET_DVR_RECTCFG_EX   struRect;
    DWORD                dwContentNum;
    BYTE                 byLedContent[MAX_LEDCONTENT_NUM];
    BYTE                 byMoveMode;
    BYTE                 byFontSize;
    BYTE                 byMoveDirection;
    BYTE                 byMoveSpeed;
    BYTE                 byRes2[124];
}NET_DVR_VIRTUALLED_PARAM,*LPNET_DVR_VIRTUALLED_PARAM;
```

Members

dwSize

结构体大小

byEnable

使能：0- 禁用，1- 启用

byDispMode

显示模式，1-透明，2-半透明，3-覆盖

byRes1

保留，置为 0

struContentColor

字体颜色

struBackColor

背景颜色

struRect

区域坐标

dwContentNum

字符的个数

byLedContent

虚拟 LED 字符内容

byMoveMode

字符移动模式：1- 普通，2- 平滑，3- 静止

byFontSize

字体大小：1- 1 倍，2- 2 倍，3- 4 倍

byMoveDirection

移动方向：0- 默认（从左到右），1- 从左到右，2- 从右到左，3- 从上到下，4- 从下到上

byMoveSpeed

移动速度：1- 速度 1，2- 速度 2

byRes2

保留，置为 0

8.190 NET_DVR_VS_INPUT_CHAN_CFG:虚拟屏输入通道配置

```
struct{
    DWORD                dwSize;
    DWORD                dwVSInputChan;
    NET\_DVR\_RES\_INFO    struResolutin;
    BYTE                byRes[64];
}NET_DVR_VS_INPUT_CHAN_CFG, *LPNET_DVR_VS_INPUT_CHAN_CFG;
```

Members

dwSize

结构体大小

dwVSInputChan

虚拟屏输入通道号

struResolutin

分辨率

byRes

保留，置为 0

Remarks

- 虚拟屏子板的输入通道初始是不存在的，需要调用 [NET_DVR_SetSTDConfig](#)(命令：
`NET_DVR_SET_VS_INPUT_CHAN_INIT_ALL`)初始化之后才会存在，因此参数都需要外部传入。对于每个通道，都需要上层下发通道号和该通道的分辨率列表，支持的通道数量和每个通道支持的分辨率列表都可以通道能力集获取到。
- 虚拟屏支持的通道数和支持的分辨率列表，对应电视墙能力集（[WallAbility](#)），相关接口：
[NET_DVR_GetDeviceAbility](#)，能力集类型：WALL_ABILITY，节点：<VirtualScreen>。

8.191 NET_DVR_VS_INPUT_CHAN_INIT:单个虚拟屏输入通道初始化参数

```
struct{
    DWORD          dwSize;
    DWORD          dwVSInputChan;
    DWORD          dwResNums;
    NET\_DVR\_RES\_INFO struResList[MAX_RES_NUM_ONE_VS_INPUT_CHAN];
    BYTE           byRes[32];
}NET_DVR_VS_INPUT_CHAN_INIT, *LPNET_DVR_VS_INPUT_CHAN_INIT;
```

Members

dwSize

结构体大小

dwVSInputChan

虚拟屏输入通道号

dwResNums

分辨率的个数

struResList

分辨率列表

byRes

保留，置为 0

Remarks

- 虚拟屏子板的输入通道初始是不存在的，需要调用 [NET_DVR_SetSTDConfig](#)(命令: `NET_DVR_SET_VS_INPUT_CHAN_INIT_ALL`)初始化之后才会存在，因此参数都需要外部传入。对于每个通道，都需要上层下发通道号和该通道的分辨率列表，支持的通道数量和每个通道支持的分辨率列表都可以通道能力集获取到。
- 虚拟屏支持的通道数和支持的分辨率列表，对应电视墙能力集（[WallAbility](#)），相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：WALL_ABILITY，节点：<VirtualScreen>。

8.192 NET_DVR_VS_INPUT_CHAN_INIT_LIST:虚拟屏输入通道初始化参数

```
struct{
    DWORD          dwSize;
    NET\_DVR\_VS\_INPUT\_CHAN\_INIT struChanList[MAX_VS_INPUT_CHAN_NUM];
    BYTE           byRes[32];
}NET_DVR_VS_INPUT_CHAN_INIT_LIST, *LPNET_DVR_VS_INPUT_CHAN_INIT_LIST;
```

Members

dwSize

结构体大小

struChanList

虚拟屏单个输入通道初始化参数，每位数组表示一个虚拟屏输入通道

byRes

保留，置为 0

Remarks

- 虚拟屏子板的输入通道初始是不存在的，需要调用 [NET_DVR_SetSTDConfig](#)(命令: `NET_DVR_SET_VS_INPUT_CHAN_INIT_ALL`)初始化之后才会存在，因此参数都需要外部传入。对于每个通道，都需要上层下发通道号和该通道的分辨率列表，支持的通道数量和每个通道支持的分辨率列表都可以通道能力集获取到。
- 虚拟屏支持的通道数和支持的分辨率列表，对应电视墙能力集（[WallAbility](#)），相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：WALL_ABILITY，节点：<VirtualScreen>。

8.193 NET_DVR_VS_NETSRC_CFG:虚拟屏网络源参数

```
struct{
    DWORD    dwSize;
    DWORD    dwVSInputChan;
    BYTE     byDispUrl[MAX_URL_LEN];
    BYTE     byRes[128];
}NET_DVR_VS_NETSRC_CFG, *LPNET_DVR_VS_NETSRC_CFG;
```

Members

dwSize

结构体大小

dwVSInputChan

虚拟屏输入通道号

byDispUrl

显示 URL，网络图片的 HTTP 访问 URL，设备通过该 URL 获取图片在虚拟屏上显示

byRes

保留，置为 0

8.194 NET_DVR_WALL_CFG:电视墙全局参数

```
struct{
    DWORD                                dwSize;
    BYTE                                byTransparency;
    BYTE                                byWinStaticMode;
    BYTE                                byStreamFailedMode;
    BYTE                                byRes1;
    NET\_DVR\_SUBSTREAM\_SWITCH\_CFG struSubStreamSwitch;
    BYTE                                byRes[52];
}NET_DVR_WALL_CFG, *LPNET_DVR_WALL_CFG;
```

Members

dwSize

结构体大小

byTransparency

透明度，取值范围：0~100，0 为不透明，对应电视墙能力集（[WallAbility](#)）中的节点<Transparency>

byWinStaticMode

窗口解码停止显示模式：1- 黑屏，2- 显示最后一帧图像，对应电视墙能力集（[WallAbility](#)）中的节点<windowStaticMode>

byStreamFailedMode

取流失败显示模式：1- 显示“无网络视频信号”，2- 显示最后一帧图像，对应电视墙能力集（[WallAbility](#)）中的节点<streamFailedMode>

byRes1

保留，置为 0

struSubStreamSwitch

子码流自动切换配置，对应电视墙能力集（[WallAbility](#)）中的节点<SubStreamAutoSwitchCap>

byRes

保留，置为 0

8.195 NET_DVR_WALL_RELATION_CFG:设备墙与物理墙关联配置

```
struct{
    DWORD    dwSize;
    BYTE      byEnable;
    BYTE      byRealWallNo;
    BYTE      byRes[14];
}NET_DVR_WALL_RELATION_CFG, *LPNET_DVR_WALL_RELATION_CFG;
```

Members

dwSize

结构体大小

byEnable

是否关联：0- 不关联，非 0- 关联

byRealWallNo

物理墙号，从 1 开始

byRes

保留，置为 0

Remarks

- 当一个物理墙上有多个控制设备时，且设备支持多墙，这时就有可能出现设备使用不同的墙号管理。使用物理墙与设备墙关联，这样就可以使用物理墙号统一管理这些设备。
- 默认设备墙与物理墙的墙号一一对应，例如物理墙 1 对应设备墙 1。

8.196 NET_DVR_WALL_WIN_STATUS:窗口状态信息

```
struct{
    DWORD    dwSize;
    BYTE      byDecodeStatus;
    BYTE      byStreamType;
```

```

BYTE    byPacketType;
BYTE    byFpsDecV;
BYTE    byFpsDecA;
BYTE    byRes1[7];
DWORD   dwDecodedV;
DWORD   dwDecodedA;
WORD    wImgW;
WORD    wImgH;
BYTE    byStreamMode;
BYTE    byRes2[31];

```

```
}NET_DVR_WALL_WIN_STATUS, *LPNET_DVR_WALL_WIN_STATUS;
```

Members

dwSize

结构体大小

byDecodeStatus

当前解码状态：0- 未启动，1- 已启动

byStreamType

编码类型，具体定义见下表：

宏定义	宏定义值	含义
NET_DVR_ENCODER_UNKOWN	0	未知编码格式
NET_DVR_ENCODER_H264	1	private 264
NET_DVR_ENCODER_S264	2	标准 H264
NET_DVR_ENCODER_MPEG4	3	MPEG4
NET_DVR_ORIGINALSTREAM	4	原始流
NET_DVR_PICTURE	5	智能分析联动的报警图片
NET_DVR_ENCODER_MJPEG	6	MJPEG
NET_DVR_ECONDER_MPEG2	7	MPEG2
NET_DVR_ENCODER_H265	8	标准 H265

byPacketType

封装格式，具体定义见下表：

宏定义	宏定义值	含义
NET_DVR_STREAM_TYPE_UNKOWN	0	未知封装格式
NET_DVR_STREAM_TYPE_PRIVT	1	私有自定义封装格式
NET_DVR_STREAM_TYPE_TS	7	TS 封装格式
NET_DVR_STREAM_TYPE_PS	8	PS 封装格式
NET_DVR_STREAM_TYPE_RTP	9	RTP 封装格式
NET_DVR_STREAM_TYPE_ORIGIN	10	未封装

byFpsDecV

视频解码帧率

byFpsDecA

音频解码帧率

byRes1

保留，置为 0

dwDecodedV

已解码的视频帧数

dwDecodedA

已解码的音频帧数

wImgW

解码器当前的图像宽度

wImgH

解码器当前的图像高度

byStreamMode

码流源(此参数对视频综合平台有效): 1-网络, 2-综合平台内部编码子系统, 0xff-无效

byRes2

保留，置为 0

8.197 NET_DVR_WALLOUTPUTPARAM:电视墙显示输出参数

```

struct{
    DWORD                dwSize;
    DWORD                dwResolution;
    NET_DVR_VIDEOEFFECT struRes;
    BYTE                 byVideoFormat;
    BYTE                 byDisplayMode;
    BYTE                 byBackgroundColor;
    BYTE                 byUseEDIDResolution;
    WORD                 wLEDWidth;
    WORD                 wLEDHeight;
    BYTE                 byRes2[56];
}NET_DVR_WALLOUTPUTPARAM, *LPNET_DVR_WALLOUTPUTPARAM;

```

Members

dwSize

结构体大小

dwResolution

输出分辨率，LED 分辨率（wLEDWidth、wLEDHeight）和 EDID 分辨率没有设置和启用时有效

struRes

视频参数，保留

byVideoFormat

视频制式：0-无效，1-N 制，2-P 制

byDisplayMode

输出连接模式（只能获取，不能设置）：1- BNC，2- VGA，3- HDMI，4- DVI，5- SDI，6- FIBER，7- RGB，8- YprPb，9- VGA/HDMI/DVI 自适应，10- 3GSDI，11- VGA/DVI 自适应，0xff- 无效

byBackgroundColor

背景色：0- 不支持背景色，1- 红，2- 绿，3- 蓝，4- 黄，5- 紫，6- 青，7- 黑，8- 白

byUseEDIDResolution

是否使用 EDID 分辨率：0- 不使用，1- 使用，LED 分辨率（wLEDWidth、wLEDHeight）没有设置时有效，启用之后设备自动获取显示输出口接的显示器的 EDID 分辨率

wLEDWidth

自定义输出分辨率宽（LED 分辨率）

wLEDHeight

自定义输出分辨率高（LED 分辨率）

byRes2

保留，置为 0

Remarks

- 区分显示输出口连接的是 LCD 屏还是 LED 屏可判断 wLEDWidth 或 wLEDHeight 是否为 0 来实现，wLEDWidth 或 wLEDHeight 为 0 则连接的是 LCD 屏，否则连接的是 LED 屏。如果设置了 LED 屏分辨率，则 dwResolution、byUseEDIDResolution 分辨率无效。
- 设备的视频输出参数能力，可以通过设备能力集获取，对应电视墙能力集([WallAbility](#))，相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：WALL_ABILITY，节点：<VoutResource>、<isSupportEDIDResolution>。

8.198 NET_DVR_WALLSCENECFG:电视墙场景信息

```
struct{
    DWORD    dwSize;
    BYTE     sSceneName[NAME_LEN];
    BYTE     byEnable;
    BYTE     bySceneIndex;
    BYTE     byRes [78];
}NET_DVR_WALLSCENECFG, *LPNET_DVR_WALLSCENECFG;
```

Members

dwSize

结构体大小

sSceneName

场景名称

byEnable

场景是否有效：0-无效，1-有效

bySceneIndex

场景号，只能获取。获取所有场景时使用该参数。

byRes

保留，置为 0

8.199 NET_DVR_WALLWIN_INFO:电视墙窗口信息

```
struct{
    DWORD    dwSize;
    DWORD    dwWinNum;
```

```

    DWORD    dwSubWinNum;
    DWORD    dwWallNo;
    BYTE      byRes[12];
}NET_DVR_WALLWIN_INFO,*LPNET_DVR_WALLWIN_INFO;

```

Members

dwSize

结构体大小

dwWinNum

窗口号(组合)

dwSubWinNum

子窗口号(组合)

dwWallNo

电视墙号(组合)

byRes

保留，置为 0

8.200 NET_DVR_WALLWINCFG:电视墙窗口参数

```

struct{
    DWORD          dwSize;
    BYTE           byEnable;
    BYTE           byRes1[7];
    DWORD          dwWinNum;
    DWORD          dwLayerIndex;
    NET\_DVR\_RECTCFG struWinPosition;
    DWORD          dwDeviceIndex;
    WORD           wInputIndex;
    BYTE           byRes2[14];
} NET_DVR_WALLWINCFG, *LPNET_DVR_WALLWINCFG;

```

Members

dwSize

结构体大小

byEnable

是否使能：0- 否，1- 是

byRes1

保留，置为 0

dwWinNum

窗口序号，从 1 开始

dwLayerIndex

窗口相对应的图层号

struWinPosition

目的窗口(相对显示墙)

dwDeviceIndex

设备序号，视频综合平台不支持，设为 0

wInputIndex

输入信号源，视频综合平台不支持，设为 0

byRes2

保留，置为 0

8.201 NET_DVR_WALLWINPARAM:电视墙窗口相关参数

```
struct{
    DWORD    dwSize;
    BYTE      byTransparency;
    BYTE      byWinMode;
    BYTE      byEnableSpartan;
    BYTE      byDecResource;
    BYTE      byRes[32];
} NET_DVR_WALLWINPARAM, *LPNET_DVR_WALLWINPARAM
```

Members

dwSize

结构体大小

byTransparency

透明度，取值范围：0~100，0 为不透明

byWinMode

窗口分屏模式，能力集获取

byEnableSpartan

畅显使能：0-关，1-开

byDecResource

为窗口分配的解码资源：1- D1， 2- 720P， 3- 1080P

byRes

保留，置为 0

Remars

设备支持配置的参数能力通过接口 [NET_DVR_GetDeviceAbility](#)（能力集类型：WALL_ABILITY）获取。

8.202 NET_DVR_WIN_LOGO_CFG:窗口 LOGO 参数

```
struct{
    DWORD    dwSize;
    DWORD    dwLogoNo;
    BYTE      byEnable;
    BYTE      byRes1[3];
    DWORD    dwCoordinateX;
    DWORD    dwCoordinateY;
    BYTE      byFlash;
    BYTE      byTranslucent;
    BYTE      byRes2[34];
}
```



```
}NET_DVR_WIN_LOGO_CFG,*LPNET_DVR_WIN_LOGO_CFG;
```

Members

dwSize

结构体大小

dwLogoNo

LOGO 号

byEnable

LOGO 是否显示：1- 显示，0- 隐藏

byRes1

保留，置为 0

dwCoordinateX

图片显示区域 X 坐标

dwCoordinateY

图片显示区域 Y 坐标

byFlash

是否闪烁：1- 闪烁，0- 不闪烁

byTranslucent

是否半透明：1- 半透明，0- 不半透明

byRes2

保留，置为 0

8.203 NET_DVR_WIN_ROAM_SWITCH_CFG:窗口漫游开关参数

```
struct{
```

```
    DWORD    dwSize;
```

```
    BYTE     byEnableRoam;
```

```
    BYTE     byRes[31];
```

```
}NET_DVR_WIN_ROAM_SWITCH_CFG,*LPNET_DVR_WIN_ROAM_SWITCH_CFG;
```

Members

dwSize

结构体大小

byEnableRoam

是否允许窗口漫游：0- 否，非 0- 是

byRes

保留，置为 0

Remarks

- 开启窗口漫游时，解码器使用本文档介绍的拼控接口进行配置，解码器显示通道分屏及解码通道操作相关接口均失效。
- 关闭窗口漫游时，遵循普通解码器接口，以显示通道和解码通道方式操作设备，电视墙漫游相关接口失效。显示通道和解码通道方式操作相关接口请参考《设备网络 SDK 编程指南（解码器）.pdf》。
- 设备是否支持窗口漫游能力，可以通过设备能力集进行判断，对应解码器能力集(DECODER_ABILITY)，相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：DECODER_ABILITY，节点：<SupportWinRoamSwitch>。

8.204 NET_DVR_WININDEX_INFO:窗口编号信息

```
struct{
    DWORD    dwWinIndex;
    DWORD    dwSubWinIndex;
    BYTE     byType;
    BYTE     byWallNo;
    BYTE     byRes[6];
}NET_DVR_WININDEX_INFO, *LPNET_DVR_WININDEX_INFO;
```

Members

dwWinIndex

窗口号

dwSubWinIndex

子窗口号

byType

资源类型：1- 解码，2- 回显

byWallNo

墙号

byRes

保留，置为 0

8.205 NET_DVR_XML_CONFIG_INPUT:透传接口输入参数

```
struct{
    DWORD    dwSize;
    void     *lpRequestUrl;
    DWORD    dwRequestUrlLen;
    void     *lpInBuffer;
    DWORD    dwInBufferSize;
    DWORD    dwRecvTimeOut;
    BYTE     byRes[32];
}NET_DVR_XML_CONFIG_INPUT, *LPNET_DVR_XML_CONFIG_INPUT;
```

Members

dwSize

结构体大小

lpRequestUrl

请求信令，字符串格式

dwRequestUrlLen

请求信令长度，字符串长度

lpInBuffer

输入参数缓冲区，XML 格式

dwInBufferSize

输入参数缓冲区大小

dwRecvTimeOut

接收超时时间，单位：ms，填 0 则使用默认超时 5s

byRes

保留，置为 0

Remarks

- 对于不同的协议功能（IpRequestUrl 输入的 URL 命令），IpInBuffer 对应不同的内容，详见 NET_DVR_STDXMLConfig 接口中"Remarks"说明。
- NET_DVR_STDXMLConfig 接口是直接透传的 ISAPI 协议命令，该结构体中的请求信令以及输入参数信息的详细内容可以参考 ISAPI 协议文档。

8.206 NET_DVR_XML_CONFIG_OUTPUT:透传接口输出参数

```
struct{
    DWORD    dwSize;
    void      *lpOutBuffer;
    DWORD    dwOutBufferSize;
    DWORD    dwReturnedXMLSize;
    void      *lpStatusBuffer;
    DWORD    dwStatusSize;
    BYTE      byRes[32];
}NET_DVR_XML_CONFIG_OUTPUT,*LPNET_DVR_XML_CONFIG_OUTPUT;
```

Members***dwSize***

结构体大小

lpOutBuffer

输出参数缓冲区，XML 格式

dwOutBufferSize

输出参数缓冲区大小(内存大小)

dwReturnedXMLSize

实际输出的 XML 内容大小

lpStatusBuffer

返回的状态参数(XML 格式: [ResponseStatus](#))，获取命令成功时不会赋值，如果不需要，可以置 NULL

dwStatusSize

状态缓冲区大小(内存大小)

byRes

保留，置为 0

Remarks

请求信令以及输入参数信息内容可以参考 ISAPI 协议文档。

8.207 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

8.208 NET_MATRIX_ANALOGMATRIX:模拟矩阵参数

```

struct{
    BYTE          bySerPortNum;
    BYTE          byMatrixSerPortType;
    BYTE          byRes1[2];
    NET\_DVR\_SINGLE\_RS232 struRS232;
    BYTE          byRes2[200];
}NET_MATRIX_ANALOGMATRIX,*LPNET_MATRIX_ANALOGMATRIX;

```

Members

bySerPortNum

连接的串口号

byMatrixSerPortType

该参数对于多屏控制器无效，置为 0

byRes1

保留，置为 0

struRS232

RS232 串口参数

byRes2

保留，置为

8.209 **NET_MATRIX_DIGITALMATRIX**:数字矩阵参数

```
struct{
    NET\_DVR\_IPADDR    struAddress;
    WORD              wPort;
    BYTE              byNicNum;
    BYTE              byRes[69];
}NET_MATRIX_DIGITALMATRIX, *LPNET_MATRIX_DIGITALMATRIX;
```

Members

struAddress

设备为数字设备时的 IP 信息

wPort

端口号

byNicNum

绑定的网口：0 - eth0，1 - eth1，考虑双网口如何通信加入绑定的网口

byRes

保留，置为 0

8.210 **NET_MATRIX_UNION**:矩阵参数联合体

```
union{
    NET\_MATRIX\_DIGITALMATRIX    struDigitalMatrix;
    NET\_MATRIX\_ANALOGMATRIX      struAnalogMatrix;
}NET_MATRIX_UNION,*LPNET_MATRIX_UNION;
```

Members

struDigitalMatrix

数字矩阵信息

struAnalogMatrix

模拟矩阵信息

8.211 电视墙能力 XML 描述

```
<!-- req, 电视墙能力集描述 -->
<WallAbility>
    <!-- req, 窗口编号 -->
    <winNum max="" min="" />
    <!--req, 场景编号-->
    <senceNum max="" min="" />
```

```

<!--req, 支持的屏幕数-->
<maxScreenNum>1</maxScreenNum>
<!--req, opt 级联综合平台编号-->
<platformNum max="" min=""/>
<!--req, opt 底图编号-->
<basePicNum max="" min=""/>
<!--req, 支持的分屏数,综合平台支持的所有分屏数,各显示输出通道实际支持的分屏数从各显示通道支持的分屏数节点获取-->
<windowMode opt="1,4,9,16,25,36">1</windowMode>
<WindowLimit> <!--req, 窗口限制-->
    <!--req, 不支持多画面分割窗口移动-->
    <NotSupportMutilWinMove>enable</NotSupportMutilWinMove>
    <!--req, 不支持跨输出的窗口分屏-->
    <NotSupportCrossDisplayNoSplit>enable</NotSupportCrossDisplayNoSplit>
</WindowLimit>
<!--req, 每个屏幕大小的基准值-->
<WindowBase>
    <windowBaseX>128</windowBaseX>
    <windowBaseY>128</windowBaseY>
</WindowBase>
<!-- req, 是否支持透明度, 0-不支持, 非 0 表示支持几层透明度-->
<Transparency><!--req, 透明度, 不支持不显示>
    <layerNum>16</layerNum><!--req, 支持几层透明度-->
    <range min="" max=""/><!-- req, 透明度设置范围-->
</Transparency>
<VoutResource>
    <videoOutNum><!--req, xs: inter, 视频输出个数--></videoOutNum>
    <audioOutNum><!--req, xs: inter, 音频输出个数--></audioOutNum>

    <AllOutputConfigOnly>
        <!--req,只支持所有显示输出口同时配置参数, 单个输出口配置不支持-->
        <enabled>true</enabled>
    </AllOutputConfigOnly>

    <SupportLEDOutputConfigAll>
        <!--opt, LED 输出口参数是否支持全墙配置-->
        <enabled>
            <!--req, xs:boolean,"true"-->
        </enabled>
    </SupportLEDOutputConfigAll>

    <supportLEDResolutionVoutType opt="hdmi,dvi,sdi"/>
    <!--opt, 支持 LED 分辨率的输出口类型, 没有该节点默认都支持, xs:string-->

```

```

<VoutResourceEntry>
    <!--req, 同级节点可以有多个-->
    <id><!--req, --></id>
    <voutType>vga<!--req, xs:string,bnc,vga,hdmi,dvi,sdi,3gsdi,vga/dvi --></voutType>
    <voutNo max="" min="" /><!--req, xs:inter,此输出类型的输出序号-->
    <videoFormat opt = "NULL,NTSC,PAL" /><!--req,支持的制式类型-->
    <outputWindowMode opt="1,4,9,16,25,36" /><!--req,显示输出口支持的窗口分屏模式-->
    <backgroundColor opt="red,green,blue,yellow,purple,cyan,black,white" /><!--req,支持的背景色-->
    <VideoEffect><!--req,支持显示效果调整-->
        <brightness min="" max="" /><!--req,支持亮度调整-->
        <contrast min="" max="" /><!--req,支持对比度调整-->
        <shariness min="" max="" /><!--req,支持锐度调整-->
        <saturation min="" max="" /><!--req,支持饱和度调整-->
        <hue min="" max="" /><!--req,支持色调调整-->
    </VideoEffect>
    <Audio>
        <enabled>true</enabled><!--req, 是否支持独立音频-->
    </Audio>

    <!--req, 输出口是否支持畅显-->
    <VoutMotionFluency>
        <enabled>true</enabled>
    </VoutMotionFluency>
    <LEDResolution>
        <enabled>true</enabled><!--req, 是否支持设置 LED 输出分辨率-->
    </LEDResolution>
    <VoutResolutionEntry><!--req, 同级节点可以有多个-->
        <resolutionName>
            <!--req, xs:string, "1080P_60HZ(1920*1080)"..., 显示输出分辨率名称-->
        </resolutionName>
        <index><!--req, xs:inter,该分辨率对应的索引值--></index>
    </VoutResolutionEntry>
</VoutResourceEntry>
</VoutResource>
<layOutNo min="" max="" /><!--req,布局序号-->
<baseMapSpan><!--req, 底图跨度,一张底图最多可覆盖的屏幕数--></baseMapSpan>
<overlayStringNum>8<!--req, 支持的字符串叠加个数--></overlayStringNum>
<inputStreamNo min="" max="" /><!--req, 输入源序号-->
<outputChanNo min="" max="" /><!--req, 输出通道序号-->
<planNo min="" max="" /><!--req, 预案序号-->
<maxPictureViewNum>4<!--req, 最大回显路数, 不支持不显示--></maxPictureViewNum>
<maxCamGroupNum><!--req, 最大分组个数, 不支持不显示--></maxCamGroupNum>
<maxNetSignalNum><!--req, 最大网络源个数, 不支持不显示--></maxNetSignalNum>
<ScreenLinkConfig><!--req, 不支持显示器连接调整不需要此节点-->

```

```

        <enabled>true</enabled><!--req, 显示器连接调整-->
    </ScreenLinkConfig>
    <StreamMedia><!--req, 不支持流媒体不需要此节点-->
        <enabled>true</enabled><!--req, 流媒体-->
    </StreamMedia>
    <ScreenControl><!--req, 不支持屏幕控制不需要此节点-->
        <enabled>true</enabled><!--req, 屏幕控制-->
    </ScreenControl>
    <maxLocalInputDeviceNum>
        <!--req, 本地编码输入设备最大个数, 不支持不显示-->
    </maxLocalInputDeviceNum>
    <maxLocalOutputDeviceNum>
        <!--req, 本地显示输出设备最大个数, 不支持不显示-->
    </maxLocalOutputDeviceNum>
    <wallNo min="" max="" /><!-- req,电视墙序号-->

    <VirtualScreen>
        <!--req,虚拟屏能力, 不支持不显示-->
        <InputChanInitCap>
            <!--opt,虚拟屏输入通道初始化能力-->
            <inputChanNums>
                <!--opt,输入通道最大数量,xs:integer-->
            </inputChanNums>
            <maxJoinResolution>
                <!--opt,最大拼接分辨率（1080P 的个数）,xs:integer-->
            </maxJoinResolution>
            <selfdefinResNums>
                <!--opt,单通道支持的自定义分辨率的最大个数,xs:integer-->
            </selfdefinResNums>
        </InputChanInitCap>
        <Resolution><!--req,虚拟屏支持的显示分辨率-->
            <ResolutionEntry>
                <!--req,同级节点可以有多个-->
                <resolutionName>
                    <!--req, xs:string, "1080P_60HZ(1920*1080)"...,显示输出分辨率名称-->
                </resolutionName>
                <index><!--req, xs:inter,该分辨率对应的索引值--></index>
            </ResolutionEntry>
        </Resolution>
    </VirtualScreen>

    <baseMapWinNo max="" min="" /><!--req, 支持的底图窗口编号范围-->
    <BaseMapPicSize>
        <!--req,支持的底图图片大小,不返回该节点的按最大支持 5120*2880,文件最大 6MB 处理-->
    </BaseMapPicSize>

```



```

    <width max="" min="" /><!--req,图片宽-->
    <height max="" min="" /><!--req,图片高-->
    <maxPicFileSize>10</maxPicFileSize><!--req,最大底图文件大小，单位：MB-->
</BaseMapPicSize>
<BaseMapWinMove>
    <!--req,是否支持底图窗口移动-->
    <enabled>true</enabled>
</BaseMapWinMove>

<PlatformPassiveDecode><!--req,综合平台是否支持被动解码-->
    <enabled>true</enabled>
</PlatformPassiveDecode>

<WallLogo><!--req,电视墙 logo，支持 logo 号的设备返回此节点-->
    <logoNo max="" min="" /><!--req,支持的 logo 编号范围，不支持 logo 编号不显示-->
    <logoSize max="" min="" /><!--req,支持的 logo 大小，以字节为单位-->
    <logoWidth max="" min="" /><!--req,logo 宽-->
    <logoHeight max="" min="" /><!--req,logo 高-->
    <logoNameLen max="" min="" /><!--req,logo 名称长度大小-->
    <logoStatus opt=" show, hide " /><!--req,logo 显示状态-->
    <logoCorordinateX max="" min="" /><!--req,logo 显示区域 X 坐标范围-->
    <logoCorordinateY max="" min="" /><!--req,logo 显示区域 Y 坐标范围-->
    <LogoFlash><!--req,支持 logo 闪烁-->
        <enabled>true</enabled>
    </LogoFlash>
    <LogoTranslucent><!--req,logo 半透明-->
        <enabled>true</enabled>
    </LogoTranslucent>
    <LogoDelete><!--req,支持 logo 删除-->
        <enabled>true</enabled>
    </LogoDelete>
</WallLogo>

<WindowLoop><!--req,支持窗口轮巡解码-->
    <maxLoopNum>12</maxLoopNum><!--req,支持的窗口轮巡解码路数-->
    <maxMonitorNum>64</maxMonitorNum><!--req,每路轮巡支持的最大监控点数-->
</WindowLoop>
<WindowDecResourceAlloc><!--req,支持窗口解码资源分配，不支持不显示-->
    <WindowDecResourceAllocEntry><!--req,支持窗口解码资源分配,同级节点可以有多个-->
        <name>D1</name><!--req,分配解码资源名称-->
        <index>1</index><!--req,索引值-->
    </WindowDecResourceAllocEntry>
</WindowDecResourceAlloc>

```

```

<MotionFluency><!--req,是否支持窗口畅显-->
    <enabled>true</enabled>
</MotionFluency>

<WinZoomStatus><!--req,支持窗口电子放大状态获取,不支持不显示-->
    <enabled>true</enabled>
</WinZoomStatus>

<WinTopAndBottom><!--req,支持窗口置顶置底-->
    <enabled>true</enabled>
</WinTopAndBottom>

<CloseAllWin><!--req,支持关闭所有窗口-->
    <enabled>true</enabled>
</CloseAllWin>

<windowStaticMode opt="black,lastframe"/>
<!--req,支持停止解码窗口显示模式配置: black-黑屏, lastframe-最后一帧-->

<LowLight opt="off,level1,level2, level3, level4, level5, level6, level7, level8"/><!--req,低照度等级-->
<NosieReduction><!--req,是否支持 3D 降噪-->
    <enabled>true</enabled>
    <autoNosieReduction>enable</autoNosieReduction>
</NosieReduction >

<Defog opt="off,level1,level2, level3, level4, level5, level6, level7"/><!--req,透雾等级-->

<VcaDecode>
<!--req,是否支持智能解码-->
    <enabled>true</enabled>
    <allWinCtrl>true</allWinCtrl><!--opt,支持所有窗口一起操作-->
</VcaDecode>

<AllWinDecSwitch><!--req,支持所有窗口解码启停-->
    <enabled>true</enabled>
</AllWinDecSwitch>

<VirtualLed>
<!--req,虚拟 led-->
    <dispMode opt="transparent, translucent,cover"/><!--req,显示模式-->
    <FontColor><!--req,字体颜色-->
        <colorY max="" min=""/>
        <colorU max="" min=""/>
        <colorV max="" min=""/>

```

```

</FontColor>
<BackgroundColor><!--req,背景颜色-->
    <colorY max="" min="" />
    <colorU max="" min="" />
    <colorV max="" min="" />
</BackgroundColor>
<characterNum max="" min="" /><!--req,字符个数-->
<moveMode opt="normal, smooth,static" /><!--req, 移动模式-->
<moveSpeed opt="speed1,speed2" /><!--req,字符移动速度-->
<fontSize opt="1times,2times,4times" /><!--req,字体大小-->
<moveDirection opt="lefttoright,righttoleft,toptobottom,bottomtotop" /><!--req,字符移动方向-->
</VirtualLed>

<ExternalMatrix><!--req,外接矩阵能力-->
    <maxMatrixNum>4</maxMatrixNum><!--req,最大外接矩阵台数-->
    <matrixNameLen max="" min="" /><!--req,外接矩阵名称长度-->
    <maxMatrixInputNum>256</maxMatrixInputNum><!--req,支持外接矩阵最大输入数-->
    <maxMatrixOutputNum>256</maxMatrixOutputNum><!--req,支持外接矩阵最大输出数-->
    <matrixChanType opt="BNC,VGA,RGB,DVI" /><!--req,支持的矩阵输出通道类型-->
    <matrixProtocol opt="ZT1.0,ZT1.0,Extron,Creator" /><!--req,支持的矩阵协议-->
    <matrixType opt="analogmatrix,digitalmatrix" /><!--req,支持的矩阵类型-->
    <DigitalMatrix><!--req,数字矩阵能力-->
        <nicNum max="" min="" /><!--req,支持的网口编号-->
    </DigitalMatrix>
    <AnalogMatrix><serPortNum max="" min="" />
        <!--req,控制器串口号范围,串口参数范围根据串口能力集-->
    </AnalogMatrix>
</ExternalMatrix>

<!--req,支持的图片回显参数选择-->
<PicViewParam>
    <picResolution opt="QCIF,CIF,D1" /><!--req,回显支持的图片分辨率-->
    <picFrameRate min="" max="" /><!--req,回显支持的图片帧率-->
</PicViewParam>

<!--req,支持显示显示通道号在电视墙上-->
<ShowDispChanNo>
    <channelType opt="DisplayChanNo,ScreenNo" /><!--req,支持显示的通道号类型-->
    <SupportDispByWallNo><!--req,支持按墙号显示-->
        <enabled>true</enabled>
    </SupportDispByWallNo>
</ShowDispChanNo>

<AudioMatrix><!--req,支持音频矩阵切换-->

```

```

    <audioChanNameLen min="" max="" /><!--req,音频通道名称长度-->
    <audioSwitchType opt="switchbyip,switchbywin" /><!--req,支持的音频切换方式-->
</AudioMatrix>

<baseMapWinNo min="" max="" /><!--req, 支持的底图窗口编号范围-->
<DownloadLogo>
    <!--req, 支持 LOGO 下载-->
    <enabled>true</enabled>
</DownloadLogo>

<JointSignal>
    <!--req,支持拼接信号源-->
    <enabled>true</enabled>
    <JointNo max="" min="" />
    <!--req,拼接编号-->
    <JointItem>
        <!--req,支持拼接的信号源类型，同级节点可以有多个-->
        <id><!--req, 信号源 ID--></id>
        <SignalType>bnc</SignalType>
        <!--req, xs:string,bnc,vga,hdmi,dvi,ip,rgb,matrix, yprpb,usb,sdi,hdi,dp,hdtvi -->
    </JointItem>
    <JointScale><!--opt,拼接规模，综合平台支持此节点返回-->
        <X86JointScale><!--opt,X86 拼接板拼接能力-->
            <maxInputNum></maxInputNum><!--opt,单个拼接最大输入数-->
            <maxHeight></maxHeight><!--opt,最大拼接高度-->
            <maxWidth></maxWidth><!--opt,最大拼接宽度-->
        </X86JointScale>
        <NormalJointScale><!--opt,普通拼接板拼接能力-->
            <maxInputNum></maxInputNum><!--opt,单个拼接最大输入数-->
            <maxHeight></maxHeight><!--opt,最大拼接高度-->
            <maxWidth></maxWidth><!--opt,最大拼接宽度-->
        </NormalJointScale>
    </JointScale>
    <DeleteJoint>
        <enabled>true</enabled><!--opt,xs:string,支持删除拼接信号源-->
    </DeleteJoint>
</JointSignal>

<SupportGetPlayingPlan>
    <!--req,支持获取正在执行的预案-->
    <enabled>true</enabled>
</SupportGetPlayingPlan>

<InputStreamV40>

```

```

<!--req, 支持本地源 V40 扩展配置-->
<ColorMode opt="SelfDefine,Sharp,Ordinary,Soft"/>
<!--req,支持色彩模式-->
</InputStreamV40>

<InputStreamCut>
  <!--req,支持信号源裁剪-->
  <CutTop max="" min=""/>
  <!--req, 上方裁剪像素值范围-->
  <CutBottom max="" min=""/>
  <!--req, 下方裁剪像素值范围-->
  <CutLeft max="" min=""/>
  <!--req, 左边裁剪像素值范围-->
  <CutRight max="" min=""/>
  <!--req, 右边裁剪像素值范围-->
</InputStreamCut>
<SupportBatchDec>
  <!--req,支持批量动态解码和批量解码开关-->
  <enabled>true</enabled>
</SupportBatchDec>
<WindowCap><!--opt-->
  <isSupportPhysicsResolutionWnd>
    <!--req,是否支持物理分辨率开窗,xs:boolean-->
  </isSupportPhysicsResolutionWnd>
  <wndWidthAlignUInt>
    <!-- dep,xs:integer,物理分辨率窗口宽度对齐单元-->
  </wndWidthAlignUInt>
  <wndHeightAlignUInt>
    <!--dep,xs:integer,物理分辨率窗口高度对齐单元-->
  </wndHeightAlignUInt>
  <isSupportDecodeOSD>
    <!--opt,xs:boolean,"true",是否支持解码 OSD 配置-->
  </isSupportDecodeOSD>
</WindowCap>
<ResetOutputDisplayPosition>
  <!--req,支持电视墙输出通道解绑-->
  <enabled><!--req,xs:boolean--></enabled>
</ResetOutputDisplayPosition>
<InputBoardCfgList>
  <!--opt,输入板参数配置能力-->
  <InputBoardCfg>
    <slotNo><!--req,xs:integer, 槽位号--></slotNo>
    <fullFrameEnable><!--req,xs:boolean,是否开启全帧率畅显--></fullFrameEnable>
  </InputBoardCfg>

```

```

</InputBoardCfgList>

<isSupportLEDArea><!--opt,xs:boolean,"true",是否支持 LED 区域--></isSupportLEDArea>

<SubStreamAutoSwitchCap>
  <!--opt,子码流自动切换-->
  <subWndWidth min="" max=""><!--req,xs:integer,子窗口宽度设定值--></subWndWidth>
  <subWndHeight min="" max=""><!--req,xs:integer,子窗口高度设定值--></subWndHeight>
</SubStreamAutoSwitchCap>

<streamFailedMode opt="noSignal,lastFrame" />
<!--opt,xs:string,窗口取流失败显示模式:noSignal-显示“无网络视频信号”,lastFrame-显示最后一帧图像-->

<isSupportWallConference>
  <!--opt,xs:boolean,"true",是否支持会议电视墙-->
</isSupportWallConference>

<isSupportBaseMapCycleSwitch>
<!--opt,xs:boolean,"true",预案是否支持底图切换(NET_DVR_SWITCH_BASEMAP)-->
</isSupportBaseMapCycleSwitch>

<isSupportEDIDResolution>
  <!--opt,xs:boolean,"true",是否支持 EDID 分辨率-->
</isSupportEDIDResolution>

</WallAbility>

```

8.212 相机拼接能力 XML 描述

输入信息

```

<CamFusionAbility>
  <!--opt,可以指定节点返回，指定节点可以是根节点的下一级节点-->
</CamFusionAbility>

```

输出信息

```

<CamFusionAbility>
  <fusionChannelNo min="1" max="6"/>
  <!--req,融合通道编号-->
  <FusionScale>
    <fusionCamNum min="2" max="6"/>
    <!--req,融合的相机数-->
    <width min="2" max="6"/>
    <!--req,融合相机列范围-->
  </FusionScale>
</CamFusionAbility>

```

```

        <height min="2" max="6"/>
        <!--req,融合相机行范围-->
    </FusionScale>
    <supportHistoryMap>enable</supportHistoryMap>
    <!--req,支持使用历史映射表融合-->
</CamFusionAbility>

```

8.213 光端机接入能力 XML 描述

输入信息

```

<OpticalDevAccessAbility>
    <!--opt,可以指定节点返回，指定节点可以是根节点的下一级节点-->
</OpticalDevAccessAbility>

```

输出信息

```

<OpticalDevAccessAbility>
    <maxNodeDevNum>
        <!--opt,xs:integer,单链路最大节点光端机数-->
    </maxNodeDevNum>
    <linkSignalCheck>
        <!--opt,xs:boolean,支持链路信号检测-->
    </linkSignalCheck>
    <linkType opt="ring,chain"/>
    <!--opt,xs:string,链路类型-->
    <SingleNodeAblity>
        <!--opt, 节点光端机设备能力-->
        <nameLen min="" max=""/>
        <!--opt,xs:integer,设备名称长度-->
        <devIDLen min="" max=""/>
        <!--opt,xs:integer,节点光端机物理 ID 长度-->
        <maxChannelNum>
            <!--opt,xs:integer,节点光端机最大通道数-->
        </maxChannelNum>
        <channelNameLen min="" max=""/>
        <!--opt,xs:integer,节点光端机通道名称长度-->
        <channelSignalCheck>
            <!--opt,xs:boolean,是否支持通道信号检测-->
        </channelSignalCheck>
        <singalType opt="SD,HD"/>
        <!--opt,xs:string,支持的信号源类型-->
    </SingleNodeAblity>
</OpticalDevAccessAbility>

```

8.214 环网能力 XML 描述

输入信息

```
<NetRingAbility>
  <subBoardNo></subBoardNo><!--req,子板号-->
</NetRingAbility>
```

输出信息

```
<NetRingAbility>
  <netRingNo min="" max=""/>
  <!--req,单子板环网号范围-->
  <protocolType opt="1,2,3"/>
  <!--req,支持的协议类型, 1-光端机组网协议(私有), 2-综合平台组网协议(私有), 3-G8032 协议(公有)-->
  <bandWidth opt="1000Mbps,1Gbps"/>
  <!--req,支持的带宽类型-->
  <supportPortAttribute opt="main,sub"/>
  <!--req,支持的端口属性-->
  <NetRingStatus>
    <ringStatus opt="connected,disconnected"/>
    <!--opt,支持的环网状态-->
    <MasterSlaveProperty>
      <enabled>true</enabled>
      <!--req,支持主从属性获取-->
    </MasterSlaveProperty>
  </NetRingStatus>
</NetRingAbility>
```

8.215 屏幕服务器登录参数配置 XML 描述

```
<ServerLoginCfgList version="2.0" xmlns="http://www.std-cgi.org/ver20/XMLSchema">
  <ServerLoginCfg/>
</ServerLoginCfgList>
```

```
<ServerLoginCfg version="2.0" xmlns="http://www.std-cgi.org/ver20/XMLSchema">
  <id><!--req,xs:integer,屏幕服务器编号--></id>
  <IpAddress><!--req, 屏幕服务器地址-->
    <ipVersion opt="v4,v6"><!--req,xs:string--></ipVersion>
    <ipAddress><!--dep,xs:string, IPV4地址--></ipAddress>
    <ipv6Address><!--dep,xs:string, IPV6地址--></ipv6Address>
  </IpAddress>
  <portNo><!--req,xs:integer,端口号--></portNo>
  <userName><!--req,xs:string,用户名--></userName>
```



```

<password><!--req,xs:string,用户名--></password>
<inputNo><!--opt,xs:integer,关联的C10S输入口编号--></inputNo>
</ServerLoginCfg>

```

8.216 输入板参数配置 XML 描述

```

<InputBoardCfg>
  <slotNo><!--req, xs:integer,槽位号--></slotNo>
  <fullFrameEnable><!--req, xs:boolean,是否开启全帧率畅显--></fullFrameEnable>
</InputBoardCfg>

```

```

<InputBoardCfgList>
  <InputBoardCfg/>
</InputBoardCfgList>

```

8.217 解码 OSD 能力 XML 描述

```

<DecodeOSD>
  <id>
    <!--req,xs:integer,子窗口号（从 1 开始）-->
  </id>
  <OSDInfoList size="" version="2.0">
    <!--req,xs:integer,子窗口解码 OSD 信息列表-->
    <OSDInfo version="2.0">
      <enabled opt="true,false">
        <!--req,xs:boolean,"true,false",是否启用-->
      </enabled>
      <flashEnabled opt="true,false">
        <!--req,xs:boolean,"true,false",是否闪烁-->
      </flashEnabled>
      <fontSize opt="large,middle,small">
        <!--req,xs:string,"large,middle,small,auto",字体大小:large-大,middle-中,small-小,auto-自适应-->
      </fontSize>
      <Color>
        <!--req,字体颜色-->
        <RGB><!--req,颜色 RGB 值, 4 个字节整数(1 字节 R+1 字节 G+1 字节 B+1 字节保留)--></RGB>
      </Color>
      <transparent min="" max="">
        <!--req,xs:integer,透明度-->
      </transparent>
      <positionX min="" max="">
        <!--req,xs:integer,OSD 左上角 X 坐标-->
      </positionX>
    </OSDInfo>
  </OSDInfoList>
</DecodeOSD>

```

```

        </positionX>
        <positionY min="" max="">
            <!--req,xs:integer,OSD 左上角 Y 坐标-->
        </positionY>
        <content max="">
            <!--req,xs:string,OSD 字符内容-->
        </content>
    </OSDInfo>
</OSDInfoList>
</DecodeOSD>

```

8.218 会议信息报警 XML 描述

```

<WallConference>
    <id><!--req,xs:string, 会议 ID--></id>
    <conferenceName><!--req,xs:string, 会议名称--></conferenceName>
    <conferenceStreamUrl><!-- req,xs:string, 会议流 URL（会议混合流） --></conferenceStreamUrl>
    <conferenceEndTime><!-- req,xs:string,"GB8601", 会议结束时间--></conferenceEndTime>
    <WallTerminalList>
        <!--req, 会议的终端列表-->
        <WallTerminal>
            <id><!--req,xs:integer, 终端 ID--></id>
            <terminalName><!--req,xs:string, 终端名称--></terminalName>
            <isChairman><!--req,xs:boolean,"true,false", 是否是主席位--></isChairman>
            <camStreamUrl><!--req,xs:string. 摄像机流 URL--></camStreamUrl>
            <isMultiStream><!--req,xs:boolean,"true,false", 该终端是否有双流--></isMultiStream>
            <otherStreamUrl><!--dep,xs:string, 另一路码流的 URL--></otherStreamUrl>
        </WallTerminal>
    </WallTerminalList>
    <mcuID><!--opt,xs:integer, MCU 编号--> </mcuID>
    <MCUAddress><!--opt, MCU 地址-->
        <addrFormatType>
            <!-- req,xs:string,"ipAddress,hostname", 地址类型: ipAddress-IP 地址, hostname-主机名称-->
        </addrFormatType>
        <hostName><!--dep,xs:string, 主机名称--></hostName>
        <IPAddress>
            <!--dep, IP 地址-->
            <ipVersion><!--req,xs:string,"v4,v6,dual", IP 地址版本--></ipVersion>
            <ipAddress><!--dep,xs:string, IPV4 地址--></ipAddress>
            <ipv6Address><!--dep,xs:string, IPV6 地址-->
            </ipv6Address>
        </IPAddress>
    </MCUAddress>

```

```
</WallConference>
```

8.219 显示输入参数能力 XML 描述

```
<InputConfigList size="">
  <InputConfig>
    <id><!-- req,xs:integer, 显示输入号（1 字节设备号+1 字节子板号+2 字节显示输入序号）--></id>
    <edidNo><!-- req,xs:integer, EDID 文件编号--></edidNo>
  </InputConfig>
</InputConfigList>
```

8.220 EDID 文件信息配置能力 XML 描述

```
<EdidFileList>
  <EdidFile>
    <id>
      <!--req,xs:integer, EDID 文件编号-->
    </id>
    <name>
      <!--req,xs:string, EDID 文件名称-->
    </name>
  </EdidFile>
</EdidFileList>
```

8.221 ResponseStatus

```
<ResponseStatus>
  <requestURL>
    <!-- req, ro,xs:string -->
  </requestURL>
  <statusCode>
    <!-- req, ro,xs:integer -->
    <!-- 0=1-OK, 2-Device Busy, 3-Device Error, 4-Invalid Operation, 5-Invalid XML Format, 6-Invalid XML
Content, 7-Reboot Required -->
  </statusCode>
  <statusString>
    <!-- req, ro,xs:string -->
  </statusString>
  <subStatusCode>
    <!-- req, ro,xs:string -->
  </subStatusCode>
```

```

    <id>
      <!--req,xs:integer-->
    </id>
  </ResponseStatus>

```

下表为 HTTP 响应信息中 statusCode、subStatusCode、statusString 的可能内容，供参考：

statusCode	subStatusCode	statusString
1	ok	indicate a successful operation is done
2	noMemory	Device doesn't have enough memory
	serviceUnavailable	service unavailable
	upgrading	upgrading
	deviceBusy	Device busy or no response
	reConnectIpc	Reconnect the video server
3	deviceError	Device hardware error
	badFlash	Operate flash error
	28181Uninitialized	28181 configuration uninitialized
4	notSupport	The device does't support this resource
	lowPrivilege	Not have enough privilege for this operation
	badAuthorization	The user has not passed the authentication
	methodNotAllowed	http method is not allowed
	notSetHdiskRedund	can't set redundancy attribute for hdd disk(system exists more than one non-operate hdd disk,and the attribution of a hdd disk is WR)
	invalidOperation	Invalid operation
5	badXmlFormat	Wrong XML format
6	badParameters	Parameters error
	badHostAddress	Wrong Host Address
	badXmlContent	Wrong XMLcontent
	badIPv4Address	Wrong IPv4 address
	badIPv6Address	Wrong IPv6 address
	conflictIPv4Address	IPV4 address conflict
	conflictIPv6Address	IPV6 address conflict
	badDomainName	Wrong Domain
	connectSreverFail	Failed to connect with Server
	conflictDomainName	Domain conflict
	badPort	Port conflict

	portError	Port error
	importErrorData	Failed to import data
	badNetMask	Wrong subnet mask
	badVersion	Version mismatching
	badDevType	Device type mismatching
	badLanguage	Language mismatching
	incorrentUserNameOrPassword	The user name or the password is incorrect
	invalidStoragePoolOfCloudServer	The storage pool of the cloud server is invalid, no configured storage pool or the storage pool ID is incorrect
	noFreeSpaceOfStoragePool	No free space for the storage pool.
7	rebootRequired	A reboot is required before the operation taking effect

8.222 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
MINOR_FANABNORMAL	0x31	视频综合平台风扇状态异常
MINOR_FANRESUME	0x32	视频综合平台风扇状态恢复正常
MINOR_SUBSYSTEM_ABNORMALREBOOT	0x33	视频综合平台 DM6467 异常重启
MINOR_MATRIX_STARTBUZZER	0x34	视频综合平台 DM6467 异常, 启动蜂鸣器
MINOR_PANEL_ABNORMAL	0x38	前面板连接异常
MINOR_PANEL_RESUME	0x39	前面板恢复正常
MINOR_SCREEN_SUBSYSTEM_ABNORMALREBOOT	0x3c	子板异常启动
MINOR_SCREEN_SUBSYSTEM_ABNORMALINSERT	0x3d	子板插入
MINOR_SCREEN_SUBSYSTEM_ABNORMALPULLOUT	0x3e	子板拔出
MINOR_SCREEN_ABNARMALTEMPERATURE	0x3f	温度异常
MINOR_SUBSYSTEM_IP_CONFLICT	0x4000	子板 IP 冲突
MINOR_SUBSYSTEM_NET_BROKEN	0x4001	子板断网
MINOR_FAN_ABNORMAL	0x4002	风扇异常
MINOR_BACKPANEL_TEMPERATURE_ABNORMAL	0x4003	背板温度异常

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	本地备份录像文件
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_SUBSYSTEMREBOOT	0xa0	视频综合平台：DM6467 正常重启
MINOR_MATRIX_STARTTRANSFERVIDEO	0xa1	视频综合平台：矩阵切换开始传输图像
MINOR_MATRIX_STOPTRANSFERVIDEO	0xa2	视频综合平台：矩阵切换停止传输图像

MINOR_REMOTE_SET_ALLSUBSYSTEM	0xa3	视频综合平台：设置所有 6467 子系统信息
MINOR_REMOTE_GET_ALLSUBSYSTEM	0xa4	视频综合平台：获取所有 6467 子系统信息
MINOR_REMOTE_SET_PLANARRAY	0xa5	视频综合平台：设置计划轮询组
MINOR_REMOTE_GET_PLANARRAY	0xa6	视频综合平台：获取计划轮询组
MINOR_MATRIX_STARTTRANSFERAUDIO	0xa7	视频综合平台：矩阵切换开始传输音频
MINOR_MATRIX_STOPTRANSFERAUDIO	0xa8	视频综合平台：矩阵切换停止传输音频
MINOR_LOGON_CODESPITTER	0xa9	视频综合平台：登陆码分器
MINOR_LOGOFF_CODESPITTER	0xaa	视频综合平台：退出码分器
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