

公路水路交通安全畅通与应急处置系统重大工程  
技术要求

---

第 10 部分：公路水路视频监控资源  
接入规范

2016 年 1 月

# 目 次

前 言.....	IV
引 言.....	1
1 范围.....	2
2 规范性引用文件.....	2
3 术语和定义.....	2
4 缩略语.....	3
5 视频监控总体架构.....	4
6 监控资源接入.....	5
7 主要技术指标要求.....	6
8 检测验证.....	7
9 接口分类.....	8
10 监控资源能力集定义.....	12
附 录 A.....	21
A.1 回调函数.....	21
A.1.1 操作回调函数.....	21
A.1.2 消息回调函数.....	21
A.1.3 告警回调函数.....	21
A.1.4 位置回调函数.....	22
A.1.5 媒体流回调函数.....	22
A.1.6 透明通道数据回调函数.....	22
A.2 通用.....	22
A.2.1 初始化环境.....	22
A.2.2 清除环境.....	23
A.2.3 设置消息回调函数.....	23
A.2.4 登录监控资源.....	23
A.2.5 登出监控资源.....	23
A.2.6 开启注册服务.....	23
A.2.7 关闭注册服务.....	23
A.2.8 获取 SRAI 能力集.....	24
A.2.9 获取监控资源能力集.....	24
A.2.10 获取 SRAI 状态.....	24
A.2.11 获取 SRAI 信息.....	24
A.2.12 获取最后操作的错误信息.....	24
A.2.13 获取日志.....	24
A.2.14 关闭日志句柄.....	25
A.2.15 逐条获取日志信息.....	25
A.3 媒体.....	25
A.3.1 开启实时预览.....	25
A.3.2 停止实时预览.....	25
A.3.3 查询图片.....	26
A.3.4 取消查询图片.....	26

A. 3. 5	下载图片.....	26
A. 3. 6	查询录像.....	26
A. 3. 7	取消查询录像.....	26
A. 3. 8	回放录像.....	27
A. 3. 9	停止回放录像.....	27
A. 3. 10	下载录像.....	27
A. 3. 11	停止下载录像.....	27
A. 3. 12	回放控制.....	27
A. 3. 13	打开语音对讲.....	28
A. 3. 14	停止语音对讲.....	28
A. 3. 15	发送语音数据.....	28
A. 3. 16	抓拍.....	28
A. 3. 17	手动开启录像.....	28
A. 3. 18	手动停止录像.....	29
A. 4	云台控制.....	29
A. 5	监控资源维护.....	29
A. 5. 1	重启监控资源.....	29
A. 5. 2	远程升级.....	29
A. 5. 3	获取远程升级的进度.....	29
A. 5. 4	获取远程升级的状态.....	30
A. 5. 5	终止远程升级.....	30
A. 6	参数配置.....	30
A. 6. 1	获取参数.....	30
A. 6. 2	设置参数.....	30
A. 7	透明通道.....	31
A. 7. 1	建立透明通道.....	31
A. 7. 2	断开透明通道.....	31
A. 7. 3	通过透明通道向监控资源串口发送数据.....	31
A. 7. 4	直接向 232 串口发送数据.....	31
A. 7. 5	直接向串口发送数据.....	32
A. 8	设置告警回调函数.....	32
A. 9	设置位置信息回调函数.....	32
附 录 B.....		34
B. 1	一般数据结构体定义.....	34
B. 2	监控资源基础信息定义.....	41
B. 3	告警定义.....	44
B. 4	监控资源参数.....	45
B. 5	接口调用参数.....	56
附 录 C.....		62
C. 1	布尔值.....	62
C. 2	一般常量.....	62
C. 3	错误代码.....	62
C. 4	行业类型.....	63
C. 5	消息类型.....	64
C. 6	操作类型.....	64
C. 7	通道类型.....	64
C. 8	参数类型.....	65
C. 9	告警类型.....	65
C. 10	视频编码类型.....	66
C. 11	音频编码类型.....	66
C. 12	帧类型.....	67
C. 13	数据类型.....	67

C. 14	云台控制类型.....	67
C. 15	回放录像控制命令.....	68
C. 16	能力集类型.....	70
附录 D	.....	71
附录 E	.....	102
E. 1	基本信息.....	102
E. 2	软硬件能力集.....	102
E. 3	云台控制能力集.....	103
E. 4	设备配置能力集.....	103

# 前 言

《公路水路交通安全畅通与应急处置系统重大工程技术要求》分为22部分：

- 第1部分：公路交通安全畅通与应急处置数据元；
- 第2部分：水路交通安全畅通与应急处置数据元；
- 第3部分：公路水路交通运行状态采集设备分类与代码；
- 第4部分：公路水路交通重大风险源分类与代码；
- 第5部分：公路水路交通应急资源分类与代码；
- 第6部分：公路水路交通上下行分类与代码；
- 第7部分：公路水路交通安全畅通与应急处置系统元数据；
- 第8部分：公路水路交通事件检测设备接口规范；
- 第9部分：公路交通流量监测设备接口规范；
- 第10部分：公路水路视频监控资源接入规范；
- 第11部分：水文气象环境监测设备接口规范；
- 第12部分：车载终端设备接口规范；
- 第13部分：公路网运行管理信息交换指标与格式；
- 第14部分：航道运行管理信息交换指标与格式；
- 第15部分：道路运输运行管理信息交换指标与格式；
- 第16部分：水路运输运行管理信息交换指标与格式；
- 第17部分：公路水路交通应急处置信息交换指标与格式；
- 第18部分：移动应急指挥平台互联互通技术规范；
- 第19部分：公路地理空间信息规范；
- 第20部分：公路水路地理空间信息拼接规范；
- 第21部分：公路交通运行与应急信息服务要求；
- 第22部分：水路交通运行与应急信息服务要求。

本部分为《公路水路交通安全畅通与应急处置系统重大工程技术要求》的第10部分。

本部分按照GB/T 1.1—2009给出的规则起草。

本部分由交通运输部科技司提出并归口。

本部分起草单位：浙江省交通信息中心、交通运输部路网监测与应急处置中心、浙江科技学院。

本部分主要起草人：韩海航、王久辉、吕梦蛟、高沛源、焦庆春、李念明、沙元庆、任磊。

# 引 言

为了保持技术要求的适用性与可操作性，各使用者在使用过程中，及时将对本部分的意见及建议函告浙江省交通信息中心，以便修订时研用。

地址：浙江省杭州市梅花碑 4 号，邮编：310009，邮箱：wangjh@zjt.gov.cn。

# 公路水路交通安全畅通与应急处置系统重大工程技术要求

## 第 10 部分：公路水路视频监控资源接入规范

### 1 范围

《公路水路交通安全畅通与应急处置系统重大工程技术要求》的本部分定义了公路水路视频监控的总体架构、监控资源接入、主要技术要求与检测验收,定义了公路水路视频监控资源接入接口的分类、定义了参数与调用要求等,定义了公路水路视频监控资源接入的监控资源能力集与配置集。

本部分适用于公路水路视频监控资源接入的开发、验证、部署与集成实施。

### 2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件,仅所注日期的版本适用于本文件。凡是不注日期的引用文件,其最新版本(包括所有的修改单)适用于本文件。

GB/T 28059 公路网图像信息管理系统 平台互联技术规范

JT/T 982-2015 港口视频监控系统联网技术要求

ISO/IEC 14496-10 信息技术 音频-视频对象编码 第 10 部分: 先进视频编码 (Information technology— Coding of audio-visual object—Part 10: Advanced Video Coding)

ITU-T G.711 话音频率的脉冲编码调制 (Pulse code modulation of voice frequencies)

### 3 术语和定义

#### 3.1

**监控资源** surveillance resources

为联网系统提供监控信息的各种设备和系统,监控信息的表现形式为图像、声音、事件数据、业务数据等。

监控资源包含了交通行业所有跟监控资源接入相关的前端视频监控资源,视频监控资源可按视频路数分为单路资源与多路资源。

#### 3.2

**监控资源接入接口** surveillance resources access interface

由监控资源提供,供使用者接入,可获取相关资源的音视频与基本信息,实现相关控制的软件接口。

#### 3.3

**监控资源接入单元** surveillance resources access unit

按照接口方式接收各类监控信息,并按规定的标准音视频编码传入视频监控平台的功能单元。

#### 3.4

**统一编码单元** unified encode unit

将数字音视频资源转换为统一标准音视频编码格式的功能单元。

### 3.5

**视频监控平台** video surveillance platform

为各级交通管理部门提供交通视频信息汇聚、监视、共享等功能的应用平台。

### 3.6

**视频交换单元** video exchange unit

能够实现将视频信息在网络中按照一定规则进行相互交换的逻辑单元。

### 3.7

**同步** synchronization

接口调用完成则操作完成，同时返回操作结果。

### 3.8

**异步** asynchronization

接口调用完成后不直接返回操作结果，操作结果由相应的回调函数返回。

### 3.9

**C 方式命名** C name

关于C语言变量和函数命名规则，一般采用帕斯卡命名法，混合使用大小写字符来构成标识符，第一个单词首字母大写。

### 3.10

**监控资源能力集** surveillance resources capability set

监控资源能力的集合，描述监控资源的功能和性能。

## 4 缩略语

下列缩略语适用于本部分。

CPU: 中央处理器 (Central Processing Unit)

IP: 网络之间互连的协议 (Internet Protocol)

JPEG: 联合图像专家小组 (Joint Photographic Experts Group)

NTP: 网络时间协议 (Network Time Protocol)

PPPoE: 点到点连接协议 (Point to Point Protocol Over Ethernet)

SDI: 数字串行接口 (Serial Digital Interface)

SNMP: 简单网络管理协议 (Simple Network Management Protocol)

SR: 监控资源 (Surveillance Resources)

SRAI: 监控资源接入接口 (Surveillance Resources Access Interface)



SRAU: 监控资源接入单元(Surveillance Resources Access Unit)  
SRCS: 监控资源能力集 (Surveillance Resources Capability Set)  
TCP: 传输控制协议(Transmission Control Protocol)  
UDP: 用户数据报协议(User Datagram Protocol )  
UEU: 统一编码单元(Unified Encode Unit)  
VEU: 视频交换单元(VideoExchange Unit)  
XML: 可扩展标记语言(eXtensible Markup Language)

5 视频监控总体架构

5.1 概述

公路水路视频监控应能实现各类视频监控资源规范性接入、分层管理、部省级联、区域调度应用的总体要求，总体架构分为监控资源接入、视频监控平台互联、监控信息交换和监控信息应用四部分，架构见图 1。

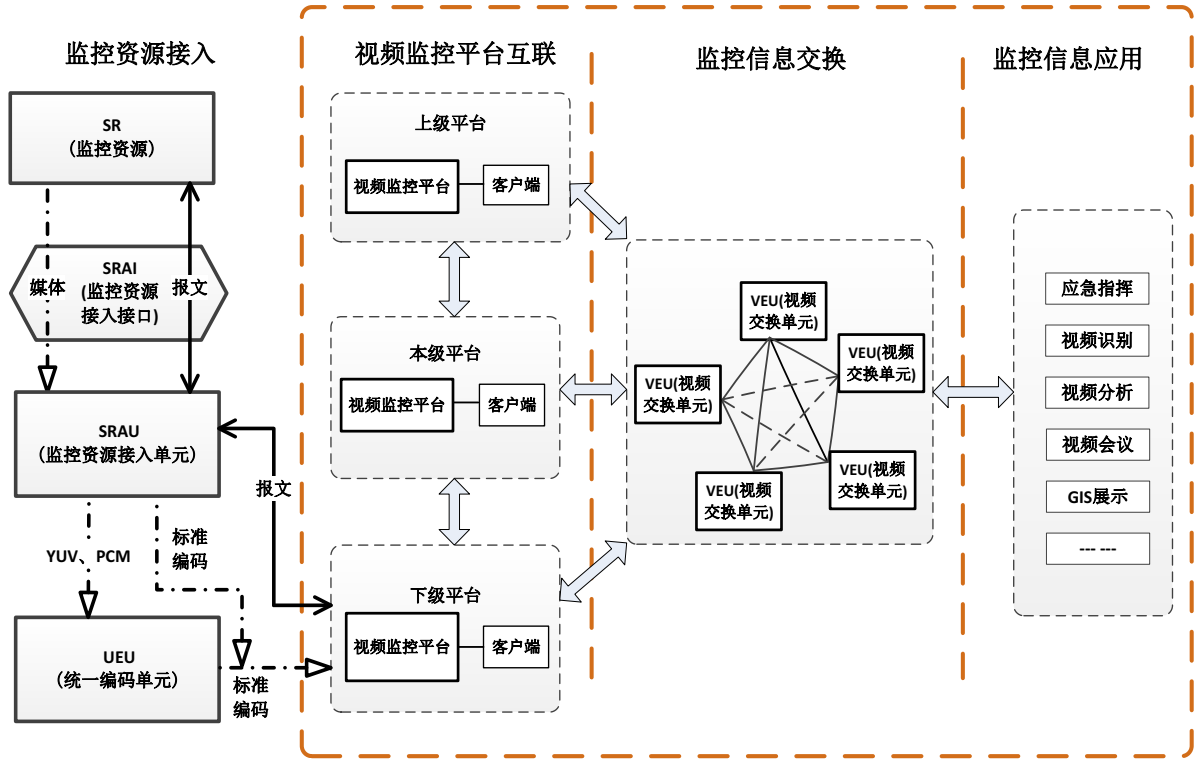


图 1 公路水路视频监控总体架构

5.2 监控资源接入

监控资源接入部分是交通视频监控的前端数据基础，通过将不同类型的监控信息进行接入，实现对相关资源的管理、控制与调用，该部分为本规范规定内容。

5.3 视频监控平台互联

视频监控平台互联部分应符合交通行业各类相关视频监控平台联网要求，其中公路视频监控平台联网应符合 GB/T 28059 的相关要求，水路视频监控平台联网应符合 JT/T 982-2015 的相关要求。

## 5.4 监控信息交换

### 5.4.1 概述

监控信息交换是为了满足交通运输管理部门对公路、水路、铁路、民航各类交通系统视频监控信息来源综合利用的需要，及满足各交通系统与其他相关部门间视频监控信息进行交换与分析应用的需求。

### 5.4.2 应用范围

监控资源交换应用于综合交通管理系统的视频交换，可实现一定地域范围内交通、公安、气象、水利等多行业之间监控视频资源的交换。

### 5.4.3 交换要求

监控资源交换过程中应包含资源类型、控制方式、视频分辨率等属性信息，并包含所属域、资源位置、资源名称、交换路径等从属信息，及质量信息、安全信息等内容，具体技术要求另行规定。

## 5.5 监控信息应用

基于交通行业特征，适时采用前后端智能识别技术，建立视频监控信息数据库，实现监控资源分析、流量检测、事件分析、视频与车辆、船舶动态之间的联动跟踪等技术，宜包含应急指挥、视频识别、视频分析、视频会议、地理信息系统展示等具体应用，具体技术要求另行规定。

## 5.6 传输链路

有线传输可采用基于 IP 网络的有线连接，包括有线专网、基于互联网的虚拟专用网络等；无线传输可采用公共运营商无线网络或卫星传输网络等方式；在接入交通内网时应具有相关的信息安全隔离与保障措施。

## 6 监控资源接入

### 6.1 监控资源

#### 6.1.1 单路资源

单路资源是提供一路监控画面的设备，一般包括：

- a) 网络摄像机（如固定枪型摄像机、云台摄像机、球型摄像机等）；
- b) 单路音视频编码器。

#### 6.1.2 多路资源

多路资源是提供多路监控画面的设备或系统，一般包括：

- a) 多路音视频编码器（支持多通道模拟设备 IP 编码，如模拟摄像机、SDI 视频流等）；
- c) 数字硬盘录像机（支持多通道模拟摄像机存储，如模拟枪机、模拟半球、模拟球机等；支持多路非压缩数字视频转压缩编码网络视频存储）；
- d) 网络硬盘录像机（支持多路网络摄像机存储）；
- e) 集中存储服务器（支持多路网络视频流存储）。

### 6.2 统一编码单元

针对各类不同监控资源的音视频格式进行统一规范，应符合整体构架的互联要求，不符合统一编码

标准规定的音视频编码格式应予以转换。

统一编码单元视频输出编码为 H. 264,应符合 ISO/IEC 14496-10 的规定;音频输出编码应为 G. 711A,应符合 ITU-T G. 711 的规定。

6.3 接入单元

6.3.1 监控信息接入方式要求

监控信息接入方式采用 SRAI 接口方式,具有通用、媒体、控制、配置等的接口功能集。

6.3.2 监控资源能力集要求

监控资源应具有能力集表述,包含的基本信息、功能信息及其相关业务信息等。

6.3.3 SRAI 输出音视频编码要求

SRAI 输出的音视频编码分别为 G. 711A 与 H. 264-ES;或用于统一编码单元输入的 YUV、PCM 信号。

7 主要技术指标要求

7.1 总体要求

7.1.1 时延抖动

接入网络时延抖动上限值为50 ms。

7.1.2 响应时间

响应时间应符合以下要求:

- a) 从 SRAU 发起视音频请求到 SRAI 视音频解码还原显示的总时延小于 500 ms;
- f) UEU 视音频编解码总时延小于 2 s;

7.1.3 SRAU 最小能力要求

SRAU最小能力要求应符合表1中的规定。

表1 接入单元最小能力要求

实现功能	操作系统	
	32 位	64 位
注册用户(资源登录)	2048 路	2048 路
实时预览	512 路	2048 路
录像回放	512 路	512 路
透明通道	512 路	512 路
语音对讲	512 路	512 路

7.2 性能要求

7.2.1 预览性能要求

在网络性能满足要求的基础上,当调用 400 路 2M 码流视频时,接入单元 CPU 资源消耗应不大于

15%;

### 7.2.2 回放性能要求

在网络性能满足要求的基础上，当回放 200 路 2M 码流视频时，接入单元 CPU 资源消耗应不大于 2%。

## 7.3 功能要求

### 7.3.1 接入模式要求

监控资源接入模式应符合本部分第 5 条的要求。

### 7.3.2 视频质量与控制要求

视频质量与控制功能应按照所接入的不同监控资源能力集予以对应，符合相关标准要求。

## 8 检测验证

### 8.1 命名原则

#### 8.1.1 接口交付包命名

接口交付包的命名方式见图 2，由多部分组成，中间以间隔符“\_”连接。

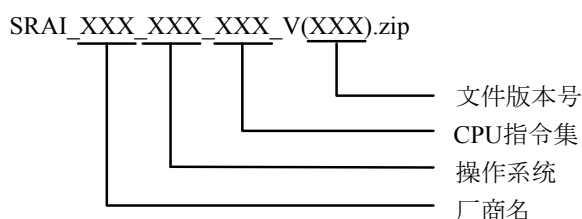


图 2 接口交付包命名格式

按照排列顺序从左至右依次为：

- 文件类型，由固定的四个大写英文字母（SRAI）构成；
- 文件所属厂商的厂商名，采用不定字长英文大写字符串表示（如 DAHUA、HIKVISION、UNIVIEW）；
- 文件适用的操作系统，采用不定字长英文小写字符串表示（如 linux、windows）；
- CPU 指令集位数，采用不定字长的字符串（如 x86、x64）；
- 文件版本号，第一个字符为英文大写字母“V”后跟英文符号“（）”，“（）”内采用不定字长、以点为间隔符的字符串（如 V(3.0.2)）；
- 文件后缀，规定为“.zip”的压缩包。

示例：SRAI\_DAHUA\_windows\_x64\_V(3.0.2).zip

#### 8.1.2 动态链接库命名

SRAI 动态链接库的命名方式见图 3，由多部分组成，中间以间隔符“\_”连接；除 SRAI 动态链接库外的其他相关文件不做命名规定。

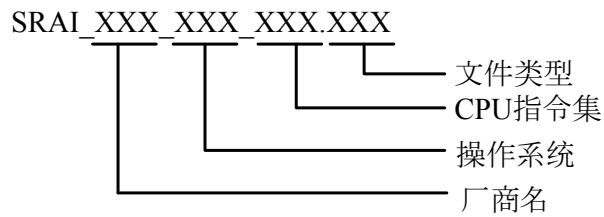


图 3 SRAI 动态链接库命名格式

按照排列顺序从左至右依次：

- a) 文件类型，由固定的四个大写英文字母（SRAI）构成；
- b) 文件所属厂商的厂商名，采用不定字长英文大写字符串表示（如 DAHUA、HIKVISION、UNIVIEW）；
- c) 文件适用的操作系统，采用不定字长英文小写字符串表示（如 linux、windows）；
- d) CPU 指令集位数，采用不定字长的字符串（如 x86、x64）；
- e) 文件后缀，采用不定字长英文小写字符串（如 dll，so）。

示例：SRAI\_DAHUA\_windows\_x64.dll

## 8.2 规范符合性验证

按照本规范对监控资源接入部分的规范符合性情况进行检测验证，验证方法的具体实施细则由相关单位提供。

对于通过验证的 SRAI 接口由相关机构鉴定唯一命名。SRAI 名称应由送检单位提供，宜结合 SRAI 开发单位的英文名称，经相关机构查证无重复，经公示后正式生效。

## 8.3 SRAI 接口及能力集验证

应对公路水路各类视频监控资源的 SRAI 接口及其能力集验证，并对接入单元性能检测，检测通过后由相关机构出具检测结果标识。

# 9 接口分类

## 9.1 概述

### 9.1.1 说明

头文件定义参见附录 D。

### 9.1.2 通用接口

通用接口主要包含初始化、设置消息回调、登录监控资源、登出监控资源、开启注册服务、日志查询等，接口名称定义见表 2。

表2 通用接口

接口	接口名称	接口	接口名称
初始化环境	SRAI_Initialize	获取 SRAI 能力集	SRAI_GetSRAICapability
清空环境	SRAI_Finalize	获取监控资源能力集	SRAI_GetResourceCapability
设置消息回调函数	SRAI_SetMsgCallback	获取 SRAI 信息	SRAI_GetSRAIInfo
登录监控资源	SRAI_Login	获取 SRAI 状态	SRAI_GetSRAIState

表 2 (续)

接口	接口名称	接口	接口名称
登出监控资源	SRAI_Logout	获取最后操作的错误信息	SRAI_GetLastError
开启注册服务	SRAI_StartServer	关闭注册服务	SRAI_StopServer
获取日志	SRAI_FindLog	关闭日志句柄	SRAI_CloseFindLog
逐条获取日志信息	SRAI_FindNextLog		

### 9.1.3 媒体接口

媒体接口主要包含开启实时预览、停止实时预览、查询图片、下载图片、查询录像文件、回放录像等，接口名称定义见表 3。

表3 媒体接口

接口	接口名称	接口	接口名称
开启实时预览	SRAI_RealPlay	停止回放录像	SRAI_StopPlayBack
停止实时预览	SRAI_StopRealPlay	打开语音对讲	SRAI_StartTalk
查询图片	SRAI_QueryPicture	停止语音对讲	SRAI_StopTalk
取消查询图片	SRAI_CancelQueryPicture	发送语音数据	SRAI_SendTalkData
下载图片	SRAI_DownloadPicture	抓拍	SRAI_Snapshot
查询录像	SRAI_QueryRecord	手动开启录像	SRAI_StartRecord
取消查询录像	SRAI_CancelQueryRecord	手动停止录像	SRAI_StopRecord
回放录像	SRAI_PlayBack	回放控制	SRAI_PlayBackControl
下载录像	SRAI_Download	停止下载录像	SRAI_StopDownload

### 9.1.4 云台控制接口

云台控制接口主要包含云台控制，接口名称定义见表 4。

表4 云台控制接口

接口	接口名称
云台控制	SRAI_PTZControl

### 9.1.5 监控资源维护接口

监控资源维护接口主要包含重启监控资源、远程升级、获取升级进度、终止远程升级等，接口名称定义见表 5。

表5 监控资源维护接口

接口	接口名称
重启监控资源	SRAI_RebootResource
远程升级	SRAI_Upgrade
获取远程升级的进度	SRAI_GetUpgradeProgress

表 5（续）

接口	接口名称
获取远程升级的状态	SRAI_GetUpgradeState
终止远程升级	SRAI_StopUpgrade

#### 9.1.6 参数配置接口

参数配置接口主要包含读取监控资源配置、设置监控资源配置，接口名称定义见表 6。

表6 参数配置接口

接口	接口名称
获取参数	SRAI_GetParam
设置参数	SRAI_SetParam

#### 9.1.7 透明通道接口

透明通道接口主要包含建立透明通道、断开透明通道、通过透明通道发送数据等，接口名称定义见表 7。

表7 透明通道接口

接口	接口名称
建立透明通道	SRAI_SerialStart
断开透明通道	SRAI_SerialStop
通过透明通道向监控资源串口发送数据	SRAI_SerialSend
直接向串口发送数据	SRAI_SerialToSerialPort
直接向 232 串口发送数据	SRAI_SerialTo232Port

#### 9.1.8 告警接口

告警接口主要包含设置告警回调，接口名称定义见表 8。

表8 告警接口

接口	接口名称
设置告警回调	SRAI_SetAlarmCallBack

#### 9.1.9 位置信息接口

位置信息接口主要包含设置位置信息回调，接口名称定义见表 9。

表9 位置信息接口

接口	接口名称
设置位置信息回调	SRAI_SetGPSCallBack

### 9.1.10 接口定义与要求

SRAI 接口定义参见附录 A，应符合 7.1.3 SRAU 最小能力要求的技术要求。

## 9.2 接口格式

### 9.2.1 数据类型

本部分使用标准数据类型见表 10。

表10 接口数据类型

类型	说明
BYTE	无符号单字节整形
BYTE[n]	n 字节长度的数组，其中元素为无符号单字节整形
CHAR	有符号单字节整形
CHAR[n]	n 字节长度的数组，其中元素为有符号单字节整形
BOOL	TRUE 或 FALSE，8 位，单字节，定义参见附录 C.1
WORD	16 位无符号整形，2 字节
DWORD	32 位无符号整形，4 字节
LONG	64 位整形，8 字节
枚举（Enumeration）	所有已知值的列表
结构体（Structure）	各种类型数据的集合，采用 4 字节对齐

### 9.2.2 接口导出

SRAI 接口导出为 \_\_stdcall（PASCAL）调用方式，接口命名为 c 方式命名。

### 9.2.3 参数

SRAI 接口中的输入输出参数由调用方分配内存，接口调用完成后由调用方释放内存。

回调函数中的参数由 SRAI 分配内存和释放内存。

SRAI 接口中的参数定义参见附录 B。

### 9.2.4 接口调用方式

接口调用按不同功能分别采用同步、异步方式实现，具体方式见表 11。

表11 接口调用方式

接口	调用方式
初始化 SRAI	同步
清空 SRAI	同步
设置消息回调函数	同步
登录监控资源	异步
登出监控资源	同步



表11（续）

接口	调用方式
开启注册服务	同步
获取 SRAI 能力集	同步
获取监控资源能力集	同步
获取 SRAI 信息	同步
获取 SRAI 状态	同步
获取最后操作的错误信息	同步
开启实时预览	异步
停止实时预览	同步
查询图片	异步
取消查询图片	同步
下载图片	同步
查询录像文件	异步
回放录像	同步
停止回放录像	同步
下载录像	同步
停止下载录像	同步
打开语音对讲	同步
停止语音对讲	同步
发送语音数据	同步
抓图	同步
关闭注册服务	同步
手动开启录像	同步
手动停止录像	同步
云台控制	同步
重启监控资源	同步
远程升级	异步
获取远程升级的进度	同步
获取远程升级的状态	同步
终止远程升级	同步
读取监控资源配置	同步

## 10 监控资源能力集定义

### 10.1 概述

#### 10.1.1 介绍

监控资源能力集采用 XML 进行描述，应符合 W3C RFC-xml，GB/T 18793 的相关要求。具体内容见附录 E。

### 10.1.2 出现次数

XML 中的出现次数规定：

- 0..1，数据元值域可填，且出现一次；
- 0..n，数据元值域可填，且可出现多次；
- 1..1，数据元值域必填，且出现一次；
- 1..n，数据元值域必填，且可出现多次。

### 10.1.3 数据类型

XML 中的数据类型和长度见表 12。

表12 类型和长度

数据类型	说明
a	字母字符
n	数字字符
an	字母数据字符
m(m=自然数)	定长m个字符（字符集默认为GB2312）
..ul	长度不确定的文本
..	从最小长度到最大长度，前面附加最小长度，后面附加最大长度
date	格式为：YYYYMMDD，YYYY表示年份，MM表示月份，DD表示日期

### 10.1.4 设备基本信息

设备基本信息主要包含设备序列号、厂商、软件版本号等，具体信息定义见表 13。

表13 设备基本信息

索引 Index	报文层 Level	XML 标记名 XML Tag Name	说明 Description	约束 Constraint	出现次数 Repeats	类型和长度 Types/Length	引用文件 Citation
1	0	BasicInformation 设备基本信息	给出设备基 本信息	M	1..1		
2	1	SerialNumber 序列号	设备序列号	M	1..1	a..128	
3	1	Factory 厂家	设备厂家	M	1..1	a..128	
4	1	SoftwareVersion 软件版本	设备软件版 本号，高 16 位是主版本， 低 16 位是次 版本	M	1..1	n	
5	1	SoftwareBuildDate 软件生成日期	设备软件生 成日期	M	1..1	date	

表 13 (续)

索引 Index	报文层 Level	XML 标记名 XML Tag Name	说明 Description	约束 Constraint	出现次数 Repeats	类型和长度 Types/Length	引用文件 Citation
6	1	DSPSoftwareVersion DSP 软件版本号	设备嵌入式 件版本号, 高 16 位是主版 本, 低 16 位 是次版本	M	1..1	n	
7	1	DSPSoftwareBuildDate DSP 软件生成日期	设备嵌入式 软件生成日 期	M	1..1	date	
8	1	PanelVersion 前面板版本	设备前面面 板版本, 高 16 位是主版本, 低 16 位是次 版本	M	1..1	n	
9	1	HardwareVersino 硬件版本	设备硬件版 本, 高 16 位 是主版本, 低 16 位是次版 本	M	1..1	n	

注: 约束条件 M 是必选, 0 是可选

#### 10.1.5 软硬件能力集

软硬件能力集主要包含设备的报警输入输出个数、设备的串口个数、音视频输入输出个数、视频预览最大路数、回放最大路数等, 具体信息定义见表 14。

表14 软硬件能力集

索引 Index	报文层 Level	XML 标记名 XML Tag Name	说明 Description	约束 Constraint	出现次数 Repeats	类型和长度 Types/Length	引用文件 Citation
1	0	BasicCapability 设备软硬件能力	给出设备软 硬件能力集	M	1..1		
2	1	HardwareCapability 设备硬件能力	给出设备硬 件能力集	M	1..1		
3	2	AlarmInPortNum 报警输入个数	设备的报警 输入的个数	M	1..1	n	
4	2	AlarmOutPortNum 报警输出个数	设备的报警 输出的个数	M	1..1	n	
5	2	RS232Num 232 串口个数	设备的 232 串 口个数	M	1..1	n	
6	2	RS485Num 485 串口个数	设备的 485 串 口个数	M	1..1	n	

表 14（续）

索引 Index	报文层 Level	XML 标记名 XML Tag Name	说明 Description	约束 Constraint	出现次数 Repeats	类型和 长度 Types/ Length	引用文件 Citation
7	2	NetworkPortNum 网络口个数	设备的网络口个数	M	1..1	n	
8	2	VideoInNum 视频输入个数	设备的视频输入个数	M	1..1	n	
9	2	AudioInNum 音频输入个数	设备的音频输入个数	M	1..1	n	
10	2	VideoOutNum 视频输出个数	设备的视频输出个数	M	1..1	n	
11	2	AudioOutNum 音频输出个数	设备的音频输出个数	M	1..1	n	
12	2	AudioTalkNum 语音对讲通道个数	设备的语音对讲通道 个数	M	1..1	n	
13	1	SoftwareCapability 软件支持能力	给出设备的软件支持 能力集	M	1..1		
14	2	ShowStringNumber 最大字符串叠加个数	设备的视频通道的最 大字符串叠加个数	M	1..1	n	
15	2	MotionDetectAlarmSupport 是否支持移动侦测	设备的视频通道是否 支持移动侦测，1-支 持，0-不支持	M	1..1	n	
16	2	VILostAlarmSupport 是否支持视频丢失报警	设备的视频通道是否 支持视频丢失报警，	M	1..1	n	
17	2	HideAlarmSupport 是否支持遮挡报警	设备的视频通道	M	1..1	n	
18	2	ShelterSupport 是否支持遮盖	设备的视频通道是否 支持遮盖	M	1..1	n	
19	2	EptzSupport 是否支持电子云台	设备的视频通道是否 支持电子云台	M	1..1	n	
20	2	PtzSupport 是否支持云台控制	设备的视频通道是否 支持云台控制	M	1..1	n	
21	2	MultipleStreamSupport 是否支持复合流	设备的视频通道是否 支持复合流	M	1..1	n	
22	2	SubStreamSupport 是否支持支持子码流	设备的视频通道是否 支持支持子码流	M	1..1	n	
23	2	BitrateLimitSupport 是否支持主子码流码率 动态限制	设备的视频通道是否 支持主子码流码率动 态限制	M	1..1	n	

表 14（续）

索引 Index	报文层 Level	XML 标记名 XML Tag Name	说明 Description	约束 Constraint	出现次数 Repeats	类型和 长度 Types/ Length	引用文件 Citation
24	2	MaxLoginNum 支持的注册最大路数	设备支持的注册最大 路数	M	1..1	n	
25	2	MaxPreviewNum 支持的预览最大路数	设备支持的预览最大 路数	M	1..1	n	
26	2	MaxPlayBackNum 支持的回放最大路数	设备支持的回放最大 路数	M	1..1	n	
27	2	RecordSupport 是否支持录像	设备是否支持录像	M	1..1	n	
28	2	PlayBackSupport 是否支持回放	设备是否支持回放	M	1..1	n	
29	2	DownloadSupport 是否支持下载录像	设备是否支持下载录 像	M	1..1	n	

#### 10.1.6 云台控制能力集

云台控制能力集主要包含巡航扫描个数、巡航速度、预置点个数、预置点名称最大长度等，具体信息见表 15。

表15 云台控制能力集

索引 Index	报文 层 Level	XML 标记名 XML Tag Name	说明 Description	约束 Constraint	出现次数 Repeats	类型和 长度 Types/ Length	引用文件 Citation
1	0	PTZAbilit 云台控制能力	给出设备云台控制能力集	M	1..1		

表 15（续）

索引 Index	报文 层 Level	XML 标记名 XML Tag Name	说明 Description	约束 Constraint	出现次 数 Repeats	类型和 长度 Types/ Length	引用文件 Citation
2	1	ControlType 云台控制	设备的视频通道云台控制，以属性方式表示，opt="light,wiper,fan,heater,aux1,aux2,zoomIn,zoomOut,focusNear,focusFar,irisOpen,irisClose,ttiltUp,tiltDown,panLeft,panRight,upLeft,upRight,downLeft,downRight,panAuto,panCircle" 分别为接通灯光电源、接通雨刷开关、接通风扇开关、接通加热器开关、接通辅助监控资源开关、接通辅助监控资源开关、焦距以速度 SS 变大、焦距以速	M	1..1	a..512	
3	1	Patrol 巡航	设备的视频通道巡航	0	0..1		
4	2	patrolNum 巡航扫描的 个数	设备的视频通道巡航扫描的个数	M	1..1	n	
5	2	presetNum 单条巡航扫描的 预置点 个数	设备的视频通道单条巡航扫描的预置点个数，以属性方式表示 属性 min 描述最小值 属性 max 描述最大值	M	1..1	n	
6	2	dwelTime 巡航点停顿 时间	设备的视频通道巡航点停顿时间，以属性方式表示 属性 min 描述最小值 属性 max 描述最大值	M	1..1	n	
7	2	Speed 巡航速度	设备的视频通道巡航速度，以属性方式表示 属性 min 描述最小值 属性 max 描述最大值	M	1..1	n	
8	2	Preset 预置点	设备的视频通道预置点	0	0..1		
9	1	PresetNum 预置点个数	设备的视频通道的预置点个数，以属性方式表示 属性 min 描述最小值 属性 max 描述最大值	M	1..1	n	

表 15（续）

索引 Index	报文层 Level	XML 标记名 XML Tag Name	说明 Description	约束 Constraint	出现次数 Repeats	类型和 长度 Types/ Length	引用文件 Citation
10	2	NameLen 预置点名称最大 长度	设备的视频通道的预置 点名称最大长度	M	1..1	n	
11	2	SpecialNo 只用于调用的预 置点列表	设备的视频通道的只用于 调用的预置点列表， 以', ' 分割，如 1, 2, 3， 不能修改和删除	M	1..1	a..128	

## 10.2 监控资源配置集定义

监控资源配置集包括 SNMP、NTP 等能力集，具体信息见表 16。

表16 网络能力集

索引 Index	报文层 Level	XML 标记名 XML Tag Name	说明 Description	约束 Constraint	出现次数 Repeats	类型和 长度 Types/ Length	引用文件 Citation
1	0	ConfigSet 配置集	给出设备配置能力集	M	1..1		
2	1	SNMPConfig SNMP 配置	SNMP 配置	M			
3	2	SNMPV1Enabled SNMP V1 开关	是否启用 SNMP V1	M	1..1	n	
4	2	SNMPV2Enabled SNMP V2 开关	是否启用 SNMP V2	M	1..1	n	
5	2	SNMPV3Enabled SNMP V3 开关	是否启用 SNMP V3	M	1..1	n	
6	2	ServerPort SNMP 消息接收端口		M	1..1	n	
7	2	ReadCommunity 读共同体		M	1..1	a..256	
8	2	WriteCommunity 写共同体		M	1..1	a..256	
9	2	TrapHostIP 自陷主机 IP 地址		M	1..1	a..125	
10	2	TrapHostPort 自陷主机端口		M	1..1	n	

表 16 (续)

索引 Index	报文层 Level	XML 标记名 XML Tag Name	说明 Description	约束 Constraint	出现次数 Repeats	类型和 长度 Types/ Length	引用文件 Citation
11	2	RWUser 读写用户		M	1..1		
12	3	UserName 用户名		M	1..1	a..125	
13	3	SecLevel 安全级别	noParity-无校验, noAuthorizationCheck-无授权校验, authorizationCheck-授权校验	M	1..1	a..125	
14	3	AuthType 认证类型	None-无加密 DES AES	M	1..1	a..125	
15	3	EncryptionType 加密类型		M	1..1	a..125	
16	2	ROUser 只读用户		M	1..1		
17	3	UserName 用户名		M	1..1	a..125	
18	3	SecLevel 安全级别	noParity-无校验, noAuthorizationCheck-无授权校验, authorizationCheck-授权校验	M	1..1	a..125	
19	3	AuthType 认证类型	None-无加密 DES AES	M	1..1	a..125	
20	3	EncryptionType 加密类型		M	1..1	a..125	
21	1	NTP		M	1..1		
22	2	Enable 开关	是否启用 NTP	M	1..1	n	
23	2	NTPServerIP NTP 服务器 IP		M	1..1	a..32	
24	2	NTPServerPort NTP 服务器端口		M	1..1	n	
25	2	Interval 校时间隔		M	1..1	n	



表 16 (续)

索引 Index	报文层 Level	XML 标记名 XML Tag Name	说明 Description	约束 Constraint	出现次数 Repeats	类型和 长度 Types/ Length	引用文件 Citation
26	1	ChannelCfg 通知配置		M	1..1		
27	2	MaxAnalogChanNum 最大模拟通道个数		M	1..1	n	
28	2	AnalogChanNum 启用的模拟通道个数		M	1..1	n	
29	2	MaxDigitalChanNum 最大数字通道个数		M	1..1	n	
30	2	DigitalChanNum 启用的数字通道个数		M	1..1	n	

**附录 A**  
**(规范性附录)**  
**接口组成和功能说明**

**A.1 回调函数**

**A.1.1 操作回调函数**

```
typedef void (__stdcall *fOptionCallBack)( DWORD dwOption
                                           , LONG lHandle
                                           , BYTE * pBuf
                                           , DWORD dwBufLen
                                           , void* pUser );
```

说明：异步调用的接口其操作结果会通过该接口注册的函数中返回

参数: dwOption[in]	操作类型，见 C.6 操作类型
lHandle[in]	操作句柄
pBuf[in]	操作结果数据，由操作类型决定
dwBufLen[in]	操作结果数据长度
pUser[in]	用户数据

返回值：无

**A.1.2 消息回调函数**

```
typedef void (__stdcall *fMessageCallBack)(DWORD dwCommand
                                           , BYTE * pBuf
                                           , DWORD dwBufLen
                                           , void* pUser);
```

说明：监控资源相关的业务消息会通过该接口注册的函数中返回

参数: dwCommand[in]	消息类型，见 C.5 消息类型
pBuf[in]	消息数据，由消息类型决定
dwBufLen[in]	消息数据长度
pUser[in]	用户数据

返回值：无

**A.1.3 告警回调函数**

```
typedef void (__stdcall *fAlarmCallBack)( DWORD dwAlarmType
                                           , BYTE * pAlarmInfo
                                           , DWORD dwAlarmInfoLen
                                           , void* pUser );
```

说明：监控资源的告警数据会通过该接口注册的函数中返回

参数: dwAlarmType[in]	告警类型，见 C.9 告警类型
pAlarmInfo[in]	告警数据，由告警类型决定
dwAlarmInfoLen[in]	告警数据长度
pUser[in]	用户数据

返回值：无

#### A. 1.4 位置回调函数

```
typedef void (__stdcall *fGPSCallBack)( LPSRAI_GPS_INFO GPS
                                         , void* pUser );
```

说明：监控资源的位置信息的数据会通过该接口注册的函数中返回

参数：GPS[in]                      位置信息  
       pUser[in]                    用户数据

返回值：无

#### A. 1.5 媒体流回调函数

```
typedef void (__stdcall *fDataCallBack)( LONG lHandle
                                         , LPSRAI_REALDATA Data
                                         , void* pUser );
```

说明：实时预览和下载的数据会通过该接口注册的函数中返回

参数：lHandle[in]                  数据句柄  
       Data[in]                    回调数据，若原始码流的视频编码类型不在标准定义内，必须回调  
                                  为解码后数据  
       pUser[in]                   用户数据

返回值：无

#### A. 1.6 透明通道数据回调函数

```
typedef void (__stdcall *fSerialDataCallBack)( LONG lHandle
                                                , BYTE * pBuf
                                                , DWORD dwBufLen
                                                , void* pUser );
```

说明：透明通道的数据会通过该接口注册的函数中返回

参数：lHandle [in]                透明通道句柄，SRAI\_SerialStart 的返回值  
       pBuf[in]                    透明通道数据  
       dwBufLen[in]                数据长度  
       pUser[in]                   用户数据

### A. 2 通用

#### A. 2.1 初始化环境

```
LONG __stdcall SRAI_Initialize ( fOptionCallBack CBK
, void* pUser );
```

说明：初始化 SRAI 环境，在其他接口之前调用

参数：CBK[in]                    操作回调函数  
       pUser[in]                   用户数据，在操作回调函数中带回

返回值：>=0                    成功  
       -1                        失败，获取错误码调用 SRAI\_GetLastError

### A.2.2 清除环境

LONG stdcall SRAI Finalize ()

说明：清除 SRAI 环境，并终止所有操作

参数：无

返回值: >=0	成功
-1	失败, 获取错误码调用 SRAI_GetLastError

### A.2.3 设置消息回调函数

```
LONG __stdcall SRAI_SetMsgCallback ( fMessageCallback CBK
                                     , void* pUser);
```

说明：设置消息回调函数，监控资源相关的业务消息会通过该接口设置的函数返回

参数: CBK[in]	消息回调函数
pUser[in]	用户数据, 在消息回调函数中带回

返回值: >=0	成功
-1	失败, 获取错误码调用 SRAI_GetLastError

#### A.2.4 登录监控资源

```
LONG    stdcall SRAI Login ( LPSRAI LOGIN INFO Info );
```

说明：登录监控资源，返回用户句柄

参数: Info [in]

返回值: >=0	成功, 返回用户句柄
-1	失败, 获取错误码调用 SRAI_GetLastError

### A.2.5 登出监控资源

```
LONG      stdcall SRAI Logout ( LONG lUserID );
```

说明：根据用户句柄登出监控资源

参数: lUserID [in] 用户句柄, SRAI Login 返回值

返回值: >=0	成功
-1	失败, 获取错误码调用 SRAI_GetLastError

### A.2.6 开启注册服务

```
LONG    stdcall SRAI StartServer ( LPSRAI START SERVER Info );
```

说明：开启注册服务供车载监控资源等移动监控资源进行注册，并返回用户句柄

参数: Info [in] 连接信息

返回值: >=0	成功, 返回用户句柄
-1	失败, 获取错误码调用 SRAI_GetLastError

### A.2.7 关闭注册服务

```
LONG      stdcall SRAI StopServer ( LONG lUserID );
```

说明：根据用户句柄关闭注册服务

参数: 1UserID [in] 用户句柄, SRAI StartServer 返回值

返回值: >=0	成功
-1	失败, 获取错误码调用 SRAI_GetLastError



```
, DWORD dwType
, LPSRAI_TIME StartTime
, LPSRAI_TIME StopTime );
```

说明：查找日志

参数：ISessionID [in]	监控资源会话 ID，SRAI_RESOURCE_INFO 中的 ISessionID
ISelectMode [in]	查询方式：0—全部；1—按类型；2—按时间；3—按时间和类型
dwType[in]	日志类型，0-类型，1-告警，2-异常，3-操作，4-日志附加信息
StartTime[in]	开始时间
StopTime[in]	结束时间
返回值：>=0	成功，返回日志查询句柄
-1	失败，获取错误码调用 SRAI_GetLastError

#### A. 2. 14 关闭日志句柄

```
LONG __stdcall SRAI_CloseFindLog ( LONG lHandle );
```

说明：关闭日志查询句柄

参数：lHandle[in]	日志查询句柄，SRAI_FindLog 返回值
返回值：>=0	成功
-1	失败，获取错误码调用 SRAI_GetLastError

#### A. 2. 15 逐条获取日志信息

```
LONG __stdcall SRAI_FindNextLog ( LONG lHandle
, LPSRAI_LOG Log );
```

说明：关闭日志查询句柄

参数：lHandle[in]	日志查询句柄，SRAI_FindLog 返回值
Log[out]	日志信息
返回值：>=0	成功
-1	失败，获取错误码调用 SRAI_GetLastError

### A. 3 媒体

#### A. 3. 1 开启实时预览

```
LONG __stdcall SRAI_RealPlay ( LONG lSessionID
, LPSRAI_REALPLAY RealPlay );
```

说明：开启实时预览，返回实时预览句柄

参数：ISessionID [in]	监控资源会话 ID，SRAI_RESOURCE_INFO 中的 ISessionID
RealPlay [in]	播放参数
返回值：>=0	成功，返回播放句柄
-1	失败，获取错误码调用 SRAI_GetLastError

#### A. 3. 2 停止实时预览

```
LONG __stdcall SRAI_StopRealPlay (LONG lHandle );
```

说明：根据实时预览句柄停止实时预览

参数：lHandle [in]	播放句柄，SRAI_RealPlay 返回值
-----------------	------------------------



说明：查询监控资源录像，结果通过操作回调函数返回

参数：lHandle [in] 查询句柄，SRAI\_QueryRecord 返回值

返回值：>=0 成功

-1 失败，获取错误码调用 SRAI\_GetLastError

### A. 3.8 回放录像

```
LONG __stdcall SRAI_PlayBack (LONG lSessionID  
                               , LPSRAI_PLAYBACKRecord);
```

说明：回放监控资源录像，并返回回放句柄

参数：lSessionID [in] 监控资源会话 ID，SRAI\_RESOURCE\_INFO 中的 lSessionID  
Record [in] 录像回放信息

返回值：>=0 成功，返回回放句柄

-1 失败，获取错误码调用 SRAI\_GetLastError

### A. 3.9 停止回放录像

```
LONG __stdcall SRAI_StopPlayBack (LONG lHandle);
```

说明：根据回放句柄停止回放录像

参数：lHandle [in] 回放句柄，SRAI\_PlayBack 返回值

返回值：>=0 成功

-1 失败，获取错误码调用 SRAI\_GetLastError

### A. 3.10 下载录像

```
LONG __stdcall SRAI_Download (LONG lSessionID  
                               , LPSRAI_PLAYBACKRecord);
```

说明：下载监控资源录像，并返回下载句柄

参数：lSessionID [in] 监控资源会话 ID，SRAI\_RESOURCE\_INFO 中的 lSessionID  
Record [in] 录像下载信息

返回值：>=0 成功，返回回放句柄

-1 失败，获取错误码调用 SRAI\_GetLastError

### A. 3.11 停止下载录像

```
LONG __stdcall SRAI_StopDownload (LONG lHandle);
```

说明：根据下载句柄停止下载录像

参数：lHandle [in] 下载句柄，SRAI\_Download 返回值

返回值：>=0 成功

-1 失败，获取错误码调用 SRAI\_GetLastError

### A. 3.12 回放控制

```
LONG __stdcall SRAI_PlayBackControl (LONG lHandle  
                                       , LPSRAI_PLAYBACK_CONTROLControl );
```

说明：根据下载句柄停止回放录像

参数：lHandle [in] 回放句柄，SRAI\_PlayBack 返回值  
Control [in] 录像回放控制参数





参数: lSessionID [in]	监控资源会话 ID, SRAI_RESOURCE_INFO 中的 lSessionID
StartRec [in]	手动录像参数
返回值: >=0	成功
-1	失败, 获取错误码调用 SRAI_GetLastError

### A. 3. 18 手动停止录像

```
LONG __stdcall SRAI_StopRecord(LONG lSessionID
                                , LPSRAI_START_RECORD StopRec);
```

说明: 手动停止监控资源录像

参数: lSessionID [in]	监控资源会话 ID, SRAI_RESOURCE_INFO 中的 lSessionID
StopRec [in]	手动录像参数
返回值: >=0	成功
-1	失败, 获取错误码调用 SRAI_GetLastError

## A. 4 云台控制

```
LONG __stdcall SRAI_PTZControl(LONG lSessionID
                                , LPSRAI_PTZ_CTRL PTZ);
```

说明: 根据控制命令不同进行不同的云台控制, 进行控制时不需开启实时预览

参数: lSessionID [in]	监控资源会话 ID, SRAI_RESOURCE_INFO 中的 lSessionID
PTZ [in]	云台控制命令
返回值: >=0	成功
-1	失败, 获取错误码调用 SRAI_GetLastError

## A. 5 监控资源维护

### A. 5. 1 重启监控资源

```
LONG __stdcall SRAI_RebootResource ( LONG lUserID);
```

说明: 重启监控资源

参数: lSessionID [in]	监控资源会话 ID, SRAI_RESOURCE_INFO 中的 lSessionID
返回值: >=0	成功
-1	失败, 获取错误码调用 SRAI_GetLastError

### A. 5. 2 远程升级

```
LONG __stdcall SRAI_Upgrade( LONG lUserID
                              , CHAR *sName);
```

说明: 使用本地文件的方式进行监控资源的远程升级, 并返回远程升级句柄

参数: lSessionID [in]	监控资源会话 ID, SRAI_RESOURCE_INFO 中的 lSessionID
sName [in]	升级的文件路径, 包括文件名
返回值: >=0	成功, 返回远程升级句柄
-1	失败, 获取错误码调用 SRAI_GetLastError

### A. 5. 3 获取远程升级的进度

LONG \_\_stdcall SRAI\_GetUpgradeProgress(LONG lHandle);

说明：根据远程升级返回的句柄获取远程升级的进度

参数：lHandle [in] 远程升级句柄，SRAI\_Upgrade 返回值

返回值：>=0 成功

-1 失败，获取错误码调用 SRAI\_GetLastError

#### A. 5. 4 获取远程升级的状态

LONG \_\_stdcall SRAI\_GetUpgradeState (LONG lHandle  
, LONG\* lState );

说明：根据远程升级返回的句柄获取远程升级的状态

参数：lHandle [in] 远程升级句柄，SRAI\_Upgrade 返回值

lState[out] 远程升级状态，定义如下：

-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

6- 升级写 flash 失败

返回值：>=0 成功

-1 失败，获取错误码调用 SRAI\_GetLastError

#### A. 5. 5 终止远程升级

LONG \_\_stdcall SRAI\_StopUpgrade(LONG lHandle );

说明：根据远程升级返回的句柄终止远程升级

参数：lHandle [in] 远程升级句柄，SRAI\_Upgrade 返回值

返回值：>=0 成功

-1 失败，获取错误码调用 SRAI\_GetLastError

### A. 6 参数配置

#### A. 6. 1 获取参数

LONG \_\_stdcall SRAI\_GetParam(LONG lSessionID  
, LPSRAI\_PARAM Param);

说明：获取监控资源参数，根据参数类型进行获取

参数：lSessionID [in] 监控资源会话 ID，SRAI\_RESOURCE\_INFO 中的 lSessionID

Param [in] 参数信息

返回值：>=0 成功

-1 失败，获取错误码调用 SRAI\_GetLastError

#### A. 6. 2 设置参数

LONG \_\_stdcall SRAI\_SetParam(LONG lSessionID  
, LPSRAI\_PARAM Param);

说明：设置监控资源参数，根据参数类型进行设置

参数：lSessionID [in]	监控资源会话 ID，SRAI_RESOURCE_INFO 中的 lSessionID
Param [in, out]	参数信息
返回值：>=0	成功
-1	失败，获取错误码调用 SRAI_GetLastError

## A. 7 透明通道

### A. 7.1 建立透明通道

```
LONG __stdcall SRAI_SerialStart( LONG lUserID
                                , LONG lSerialPort
                                , fSerialDataCallBack CBK
                                , void* pUser);
```

说明：建立和监控资源间的透明通道并返回通道句柄

参数：lSessionID [in]	监控资源会话 ID，SRAI_RESOURCE_INFO 中的 lSessionID
lSerialPort [in]	串口号：1—232 串口；2—485 串口
CBK [in]	回调函数
pUser [in]	用户数据
返回值：>=0	成功，返回透明通道句柄
-1	失败，获取错误码调用 SRAI_GetLastError

### A. 7.2 断开透明通道

```
LONG __stdcall SRAI_SerialStop( LONG lHandle);
```

说明：根据建立透明通道时返回的句柄断开透明通道

参数：lHandle [in]	透明通道句柄，SRAI_SerialStart 返回值
返回值：>=0	成功
-1	失败，获取错误码调用 SRAI_GetLastError

### A. 7.3 通过透明通道向监控资源串口发送数据

```
LONG __stdcall SRAI_SerialSend( LONG lHandle
                                , WORD wChannel
                                , BYTE *pSendBuf
                                , DWORD dwBufSize);
```

说明：根据建立透明通道时返回的句柄向监控资源的串口发送数据

参数：lHandle [in]	透明通道句柄，SRAI_SerialStart 返回值
wChannel[in]	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
pSendBuf[in]	发送数据的缓冲区指针
dwBufSize[in]	缓冲区的大小，最多 1016 字节
返回值：>=0	成功
-1	失败，获取错误码调用 SRAI_GetLastError

### A. 7.4 直接向232串口发送数据

```

LONG __stdcall SRAI_SendTo232Port ( LONG lUserID
                                     , BYTE *pSendBuf
                                     , DWORD dwBufSize);

```

说明：直接向监控资源的 232 串口发送数据

参数：lSessionID [in]                      监控资源会话 ID，SRAI\_RESOURCE\_INFO 中的 lSessionID  
       pSendBuf[in]                        发送数据的缓冲区指针  
       dwBufSize[in]                      缓冲区的大小，最多 1016 字节  
 返回值：>=0                            成功  
       -1                                失败，获取错误码调用 SRAI\_GetLastError

#### A. 7. 5 直接向串口发送数据

```

LONG __stdcall SRAI_SendToSerialPort ( LONG lUserID
                                       , DWORD dwSerialPort
                                       , DWORD dwSerialIndex
                                       , BYTE *pSendBuf
                                       , DWORD dwBufSize);

```

说明：直接向监控资源的串口发送数据

参数：lSessionID [in]                      监控资源会话 ID，SRAI\_RESOURCE\_INFO 中的 lSessionID  
       dwSerialPort[in]                    串口类型：1-232，2-485  
       dwSerialIndex[in]                  表示第几个 232 或者 485，从 1 开始  
       pSendBuf[in]                       发送数据的缓冲区指针  
       dwBufSize[in]                      缓冲区的大小，最多 1016 字节  
 返回值：>=0                            成功  
       -1                                失败，获取错误码调用 SRAI\_GetLastError

#### A. 8 设置告警回调函数

```

LONG __stdcall SRAI_SetAlarmCallBack(fAlarmCallBack CBK
                                     , void* pUser);

```

说明：调用该接口后，所有监控资源产生的告警信息通过回调函数返回，默认为全布防，取消设置为撤防

参数：CBK[in]                            告警回调函数，为 0 表示取消设置  
       pUser[in]                        用户数据，在告警回调函数中带回  
 返回值：>=0                            成功  
       -1                                失败，获取错误码调用 SRAI\_GetLastError

#### A. 9 设置位置信息回调函数

```

LONG __stdcall SRAI_SetGPSCallBack(fGPSCallBack CBK
                                    , void* pUser);

```

说明：调用该接口后，所有监控资源的位置信息通过回调函数返回

参数：CBK[in]                            位置信息回调函数，为 0 表示取消设置  
       pUser[in]                        用户数据，在位置信息回调函数中带回

返回值:  $\geq 0$

-1

成功

失败, 获取错误码调用 `SRAI_GetLastError`

## 附录 B

### （规范性附录）

#### 数据结构定义和说明

##### B.1 一般数据结构体定义

B.1.1 SRAI 信息见表 B.1。

表 B.1 SRAI 信息

字段名称	含义
sFactory	厂商名称
dwVersion	版本信息，格式为 0xAAAABBBB，其中 AAAA 为主版本号，BBBB 为次版本号
sInfo	对应监控资源系列号

```
typedef struct
{
    CHAR        sFactory[NAME_LEN];
    DWORD       dwVersion;
    CHAR        sSeries[SERIES_LEN];
}SRAI_INFO, *LPSRAI_INFO;
```

B.1.2 SRAI 能力集见表 B.2。

表 B.2 SRAI 能力集

字段名称	含义
dwLogin	最大同时登录监控资源数
dwRealPlay	最大同时实时播放数
dwPlayBack	最大同时回放数
dwUpgrade	最大同时同监控资源升级数
dwTalk	最大同时语音对讲数
dwSerial	最大同时透明通道数

```
typedef struct
{
    DWORD       dwLogin;
    DWORD       dwRealplay;
    DWORD       dwPlayback;
    DWORD       dwUpgrade;
    DWORD       dwTalk;
    DWORD       dwSerial;
}SRAI_ABILITY, *LPSRAI_ABILITY;
```

B. 1. 3 SRAI 状态见表 B. 3。

表 B. 3 SRAI 状态

字段名称	含义
dwTotalLoginNum	当前登录监控资源数
dwTotalRealPlayNum	当前实时播放数
dwTotalPlayBackNum	当前回放数
dwTotalUpgradeNum	当前同监控资源升级数
dwTotalTalkNum	当前语音对讲数
dwTotalSerialNum	当前透明通道数

```
typedef struct
{
    DWORD    dwTotalLoginNum;
    DWORD    dwTotalRealPlayNum;
    DWORD    dwTotalPlayBackNum;
    DWORD    dwTotalSerialNum;
    DWORD    dwTotalUpgradeNum;
    DWORD    dwTotalTalkNum;
}SRAI_SRAISTATE, *LPSRAI_SRAISTATE;
```

B. 1. 4 错误信息见表 B. 4。

表 B. 4 错误信息

字段名称	含义
dwError	错误代码
ErrorMsg	错误详细信息

```
typedef struct
{
    DWORD    dwError;
    CHAR      sErrorMsg[ERROR_MSG_LEN];
}SRAI_ERROR_INFO, *LPSRAI_ERROR_INFO;
```

B. 1. 5 时间见表 B. 5。

表 B. 5 时间

字段名称	含义
dwYear	年，范围 0-9999
dwMonth	月，范围 1-12
dwDay	日，范围 1-31
dwHour	时，范围 0-23
dwMinute	分，范围 0-59



表 B. 5 (续)

字段名称	含义
dwSecond	秒, 范围 0-59
dwMillisecond	毫秒, 范围 0-999

```
typedef struct
{
    DWORD    dwYear;
    DWORD    dwMonth;
    DWORD    dwDay;
    DWORD    dwHour;
    DWORD    dwMinute;
    DWORD    dwSecond;
    DWORD    dwMillisecond;
}SRAI_TIME, *LPSRAI_TIME;
```

B. 1. 6 IP 地址见表 B. 6。

表 B. 6 IP 地址

字段名称	含义
dwType	IP 地址类型, 0-IPv4, 1-IPv6
byRes	保留
sIPv4	IPv4 地址
byIPv6	IPv6 地址

```
typedef struct
{
    BYTE    byType;
    BYTE    byRes[3];
    CHAR    sIPv4[16];
    BYTE    byIPv6[128];
}SRAI_IPADDR, *LPSRAIIPADDR;
```

B. 1. 7 时间段见表 B. 7。

表 B. 7 时间段

字段名称	含义
byStartHour	开始时间: 小时
byStartMin	开始时间: 分钟
byStartSecond	开始时间: 秒
byStopHour	结束时间: 小时
byStopMin	结束时间: 分钟
byStopSecond	结束时间: 秒

```
typedef struct
{
    BYTE        byStartHour;
    BYTE        byStartMin;
    BYTE        byStopHour;
    BYTE        byStopMin;
    BYTE        byStopSecond;
}SRAI_SCHEDTIME, *LPSRAI_SCHEDTIME;
```

B. 1. 8 时间点见表 B. 8。

表 B. 8 时间点

字段名称	含义
dwMonth	月 0-11 表示 1-12 个月
dwWeekNo	0—第 1 周 1—第 2 周 2—第 3 周 3—第 4 周 4—最后一周
dwWeekDate	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

```
typedef struct
{
    DWORD        dwMonth;
    DWORD        dwWeekNo;
    DWORD        dwWeekDate;
    DWORD        dwHour;
    DWORD        dwMin;
}SRAI_TIMEPOINT, *LPSRAI_TIMEPOINT;
```

B. 1. 9 帧信息见表 B. 9。

表 B. 9 帧信息

字段名称	含义
wWidth	画面宽，单位像素。如果是音频数据则为音频声道数
wHeight	画面高。如果是音频数据则为样位率
dwStamp	时间戳，单位毫秒
wType	帧类型
wFrameRate	编码时产生的图像帧率，如果是音频数据则为采样率
dwFrameNum	帧号

```
typedef struct{
    WORD        wWidth;
    WORD        wHeight;
    DWORD        dwStamp;
```

```

WORD      wType;
WORD      nFrameRate;
DWORD     dwFrameNum;
}SRAI_FRAME_INFO;

```

B. 1. 10 图片信息见表 B. 10。

表 B. 10 图片信息

字段名称	含义
wChannel	视频通道号
sName	图片名称

```

typedef struct
{
    WORD      wChannel;
    CHAR      sName[NAME_LEN];
}SRAI_PICTURE, *LPSRAI_PICTURE;

```

B. 1. 10 图片组见表 B. 11。

表 B. 11 图片组

字段名称	含义
dwCount	图片数量
Pictures	图片列表

```

typedef struct
{
    DWORD     dwCount;
    LPSRAI_PICTURE Pictures;
}SRAI_PICTURE_GROUP, *LPSRAI_PICTURE_GROUP;

```

B. 1. 12 JPEG 图像参数见表 B. 12。

表 B. 12 JPEG 图像参数

字段名称	含义
dwPicWidth	图片宽度
dwPicLength	图片高度

表 B. 12 (续)

字段名称	含义
dwPicQuality	图片质量系数 0- 最好, 压缩因子为 100% 1- 较好, 压缩因子为 80% 2- 好, 压缩因子为 60% 3- 一般, 压缩因子为 50% 4- 较差, 压缩因子为 30% 5- 最差, 压缩因子为 10%

```
typedef struct
{
    DWORD    dwPicWidth;
    DWORD    dwPicLength;
    DWORD    dwPicQuality;
}SRAI_JPEGPARA, *LPSRAI_JPEGPARA;
```

B. 1. 13 录像信息见表 B. 13。

表 B. 13 录像信息

字段名称	含义
wChannel	视频通道号
byType	录像方式, 0-时间方式, 1-文件方式
byRes	保留
sName	文件方式, 录像名称, 当使用时间方式时该字段无效
dwTotalSeconds	文件方式, 录像文件总的时长, 单位秒
StartTime	时间方式, 开始时间, 当使用文件方式时该字段无效
EndTime	时间方式, 结束时间, 当使用文件方式时该字段无效

```
typedef struct
{
    WORD        wChannel;
    BYTE        byType;
    BYTE        byRes;
    CHAR        sName[NAME_LEN];
    DWORD       dwTotalSeconds;
    SRAI_TIME   StartTime;
    SRAI_TIME   EndTime;
}SRAI_RECORD, *LPSRAI_RECORD;
```

B. 1. 14 录像组见表 B. 14。

表 B. 14 录像组

字段名称	含义
dwCount	录像数量
Records	录像列表

```
typedef struct
{
    DWORD          dwCount;
    LPSRAI_RECORD  Records;
}SRAI_RECORD_GROUP, *LPSRAI_RECORD_GROUP;
```

B. 1. 15 位置信息见表 B. 15。

表 B. 15 位置信息

字段名称	含义
SampleTime	采样时间
dwLongitude	经度，以度为单位的经度值乘以 10 的 6 次方，精确到百万分之一度
dwLatitude	纬度，以度为单位的纬度值乘以 10 的 6 次方，精确到百万分之一度
byType	坐标系类型，0-WGS-84 坐标系，1-CGCS2000 坐标系
byRes	保留
dwDirection	方向，0-359，正北为 0，顺时针
dwSpeed	速度，单位为 1m/h
dwHeight	海拔高度，单位为米(m)

```
typedef struct
{
    SRAI_TIME  SampleTime;
    DWORD      dwLongitude;
    DWORD      dwLatitude;
    BYTE        byType;
    BYTE        byRes[3];
    DWORD      dwDirection;
    DWORD      dwSpeed;
    DWORD      dwHeight;
}SRAI_GPS_INFO, *LPSRAI_GPS_INFO;
```

B. 1. 16 回调数据见表 B. 16。

表 B. 16 回调数据

字段名称	含义
dwDataType	数据类型，回调原始数据时有效

表 B. 16（续）

字段名称	含义
FrameInfo	帧信息，回调解码后数据时有效
dwRecvSize	原始数据帧的 IP 报文长度，单位字节
dwBufSize	数据长度
pBuffer	数据

```
typedef struct
{
    DWORD          dwDataType;
    SRAI_FRAME_INFO FrameInfo;
    DWORD          dwRecvSize;
    DWORD          dwBufSize;
    BYTE           *pBuffer;
}SRAI_REALDATA, *LPSRAI_REALDATA;
```

## B. 2 监控资源基础信息定义

B. 2. 1 监控资源基本配置见表 B. 17。

表 B. 17 监控资源基本配置

字段名称	含义
sSerialNumber	序列号
dwSoftwareVersion	软件版本号，高 16 位是主版本，低 16 位是次版本
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
byAlarmInNum	告警输入个数
byAlarmOutNum	告警输出个数
byRS232Num	232 串口个数
byRS485Num	485 串口个数
byNetworkPortNum	网络口个数
byDiskCtrlNum	硬盘控制器个数
byDiskNum	硬盘个数
byVideoChanNum	视频通道个数
byDecordChans	解码路数
byVGANum	VGA 口的个数
byUSBNum	USB 口的个数
byAuxoutNum	辅口的个数

表 B. 17（续）

字段名称	含义
byAudioNum	语音口的个数
byIPChanNum	数字通道数
byRes	保留

```
typedef struct
{
    CHAR        sSerialNumber[SERIALNO_LEN];
    DWORD       dwSoftwareVersion;
    DWORD       dwSoftwareBuildDate;
    DWORD       dwDSPSoftwareVersion;
    DWORD       dwDSPSoftwareBuildDate;
    DWORD       dwPanelVersion;
    DWORD       dwHardwareVersion;
    BYTE        byAlarmInNum;
    BYTE        byAlarmOutNum;
    BYTE        byRS232Num;
    BYTE        byRS485Num;
    BYTE        byNetworkPortNum;
    BYTE        byDiskCtrlNum;
    BYTE        byDiskNum;
    BYTE        byVideoChanNum;
    BYTE        byDecordChans;
    BYTE        byVGANum;
    BYTE        byUSBNum;
    BYTE        byAuxoutNum;
    BYTE        byAudioNum;
    BYTE        byIPChanNum;
    BYTE        byRes[2];
}SRAI_RESOURCECFG, *LPSRAI_RESOURCECFG;
```

B. 2. 2 监控资源能力集见表 B. 18。

表 B. 18 监控资源能力集

字段名称	含义
dwOutLength	输出缓冲区的长度
pOutBuf	输出缓冲区指针，见监控资源能力集定义

```
typedef struct
{
    DWORD       dwOutLength;
    BYTE        *pOutBuf;
```

}SRAI\_RESOURCE\_ABILITY,\*LPSRAI\_RESOURCE\_ABILITY;

B. 2. 3 通道信息见表 B. 19。

表 B. 19 通道信息

字段名称	含义
sChanName	通道名称
wChannel	通道号，只读
byType	通道类型，只读
byStreamCount	提供的码流数量，如只有主码流则为 1，有 1 路子码流则为 2，以此类推，视频通道有效，只读

typedef struct

```
{
    CHAR        sChanName[NAME_LEN];
    WORD        wChannel;
    BYTE        byType;
    BYTE        byStreamCount;
}SRAI_CHAN_INFO,*LPSRAI_CHAN_INFO;
```

B. 2. 4 监控资源信息见表 B. 20。

表 B. 20 监控资源信息

字段名称	含义	备注
lSessionID	监控资源会话 ID	在 SRAI 会话期间唯一标示该监控资源的编号-
sName	监控资源名称	-
sResourceID	监控资源编号，只读	监控资源唯一标识
RESOURCECfg	基本配置，只读	-
byOnlineStatus	是否在线，只读	-
wFactory	监控资源厂家，只读	-
sSeriesName	监控资源系列，只读	-
byBusType	行业类型	-
sArea	所属行政区域名称	-
byGPSEnable	是否有位置信息，0-无，1-有	-
byRes1	保留	-
GPS	位置信息	可选
dwChanNum	所有通道数量，只读	-
Channels	通道列表	-

typedef struct {

```
    LONG        lSessionID;
    CHAR        sName[NAME_LEN];
    CHAR        sResourceID[NAME_LEN];
    SRAI_IPADDR RESOURCEIP;
```



```

SRAI_RESOURCECFG      RESOURCECfg;
WORD                  wFactory;
BYTE                  byOnlineStatus;
BYTE                  byBusType;
CHAR                  sSeriesName[NAME_LEN];
CHAR                  sArea[NAME_LEN];
BYTE                  byGPSEnable;
BYTE                  byRes1[3];
SRAI_GPS_INFO          GPS;
DWORD                 dwChanNum;
LPSRAI_CHAN_INFO *     Channels;
}SRAI_RESOURCE_INFO, *LPSRAI_RESOURCE_INFO;

```

### B. 3 告警定义

B. 3. 1 告警处理见表 B. 21。

表 B. 21 告警处理

字段名称	含义
dwHandleType	处理方式, 处理方式的“或”结果 0x00000000: 无响应 0x00000001: 声音警告 0x00000002: 触发告警输出 0x00000004: 触发抓图 0x00000008: 无线声光告警器联动 从低位到高位第 1 到 16 位系统保留, 其他位扩展
byRelAlarmOut	告警触发的输出通道, 告警触发的输出, 为 1 表示触发该输出

```

typedef struct
{
    DWORD          dwHandleType;
    BYTE           byRelAlarmOut[MAX_ALARMOUT];
}SRAI_HANDLEALARM, *LPSRAI_HANDLEALARM;

```

B. 3. 2 告警信息见表 B. 22。

表 B. 22 告警信息

字段名称	含义
AlarmTime	告警触发时间
dwAlarmType	告警类型
byReset	告警恢复标识 对于持续性的告警, 0-告警触发中, 1-告警恢复 对于触发式告警该字段无效

表 B. 22 (续)

字段名称	含义
byRes	保留
wChannel	告警触发的通道号或硬盘编号，由告警类型决定
PictureGroup	告警触发的图片信息
RecordGroup	告警触发的录像信息

```
typedef struct
{
    SRAI_TIME           AlarmTime;
    DWORD               dwAlarmType;
    BYTE                byReset;
    BYTE                byRes;
    WORD                wChannel;
    LPSRAI_PICTURE_GROUP PictureGroup;
    LPSRAI_RECORD_GROUP RecordGroup;
}SRAI_ALARM_INFO, *LPSRAI_ALARM_INFO;
```

#### B. 4 监控资源参数

B. 4. 1 PPPoE 参数见表 B. 23。

表 B. 23 PPPoE 参数

字段名称	含义
dwPPPOE	0-不启用, 1-启用
sPPPoEUser	PPPoE 用户名
sPPPoEPassword	PPPoE 密码
struPPPoEIP	PPPoE IP 地址

```
typedef struct
{
    DWORD           dwPPPOE;
    CHAR            sPPPoEUser[NAME_LEN];
    CHAR            sPPPoEPassword[PASSWD_LEN];
    SRAI_IPADDR     struPPPoEIP;
}SRAI_PPPOECFG, *LPSRAI_PPPOECFG;
```

B. 4. 2 网卡参数见表 B. 24。

表 B. 24 网卡参数

字段名称	含义
struIP	IP 地址

表 B. 24 (续)

字段名称	含义
struIPMask	IP 地址掩码
wMTU	MTU 设置, 默认 1500
byMACAddr	物理地址
byUseDhcp	是否启用 DHCP 0xFF-无效 0-不启用 1-启用
byRes	保留
struGatewayIpAddr	网关地址

```
typedef struct
{
    SRAI_IPADDR    struIP;
    SRAI_IPADDR    struIPMask;
    WORD           wMTU;
    BYTE           byUseDhcp;
    BYTE           byRes;
    BYTE           byMACAddr[MACADDR_LEN];
    SRAI_IPADDR    struGatewayIpAddr;
}SRAI_ETHERNET, *LPSRAI_ETHERNET;
```

B. 4. 3 网络参数见表 B. 25。

表 B. 25 网络参数

字段名称	含义
byDefaultRoute	默认路由, 0 表示 struEtherNet[0], 1 表示 struEtherNet[1]
byNetworkCardNum	监控资源实际可配置的网卡数目, 只读
byRes	保留
struEtherNet	以太网口
byIpResolver	IP 解析服务器域名或 IP 地址
wIpResolverPort	IP 解析服务器端口号
wLocalPort	通讯端口
wHttpPortNo	HTTP 端口号
byRes1	保留
struMulticastIpAddr	多播组地址
struPPPoE	PPPoE 配置

```
typedef struct
{
    BYTE           byDefaultRoute;
    BYTE           byNetworkCardNum;
    BYTE           byRes[2];
    SRAI_ETHERNET struEtherNet[MAX_NETWORK_CARD];
    BYTE           byIpResolver[MAX_DOMAIN_NAME];
```

```

WORD          wIpResolverPort;
WORD          wLocalPort;
WORD          wHttpPortNo;
BYTE          byRes1[2];
SRAI_IPADDR   struMulticastIpAddr;
SRAI_PPPOECFG struPPPoE;
}SRAI_NETCFG, *LPSRAI_NETCFG;

```

B. 4. 4 视频区域见表 B. 26。

表 B. 26 视频区域

字段名称	含义
wHideAreaTopLeftX	视频区域的 x 坐标
wHideAreaTopLeftY	视频区域的 y 坐标
wHideAreaWidth	视频区域的宽
wHideAreaHeight	视频区域的高

```

typedef struct
{
    WORD      wHideAreaTopLeftX;
    WORD      wHideAreaTopLeftY;
    WORD      wHideAreaWidth;
    WORD      wHideAreaHeight;
}SRAI_SHELTER, *LPSRAI_SHELTER;

```

B. 4. 5 信号丢失告警参数见表 B. 27。

表 B. 27 信号丢失告警参数

字段名称	含义
byEnableHandleVILost	是否处理信号丢失告警
byRes	保留
strVILostHandleType	处理方式
struAlarmTime	布防时间

```

typedef struct
{
    BYTE          byEnableHandleVILost;
    BYTE          byRes[3];
    SRAI_HANDLEALARM strVILostHandleType;
    SRAI_SCHEDTIME struAlarmTime[MAX_DAYS][8];
}SRAI_VIDEO_LOST, *LPSRAI_VIDEO_LOST;

```

B. 4. 6 视频遮盖参数见表 B. 28。

表 B. 28 视频遮盖参数

字段名称	含义
dwEnableHideAlarm	是否启动遮挡告警, 0-否, 1-低灵敏度 2-中灵敏度 3-高灵敏度
wHideAlarmAreaTopLeftX	遮挡区域的 x 坐标
wHideAlarmAreaTopLeftY	遮挡区域的 y 坐标
wHideAlarmAreaWidth	遮挡区域的宽
wHideAlarmAreaHeight	遮挡区域的高
strHideAlarmHandleType	处理方式
struAlarmTime	布防时间

```
typedef struct
```

```
{
    DWORD          dwEnableHideAlarm;
    WORD           wHideAlarmAreaTopLeftX;
    WORD           wHideAlarmAreaTopLeftY;
    WORD           wHideAlarmAreaWidth;
    WORD           wHideAlarmAreaHeight;
    SRAI_HANDLEALARM strHideAlarmHandleType;
    SRAI_SCHEDTIME struAlarmTime[MAX_DAYS][8];
}SRAI_VIDEO_HIDE, *LPSRAI_VIDEO_HIDE;
```

B. 4. 7 移动侦测见表 B. 29。

表 B. 29 移动侦测

字段名称	含义
byMotionScope	侦测区域, 0-96 位, 表示 64 行, 共有 96*64 个小宏块, 为 1 表示是移动侦测区域, 0-表示不是
byMotionSensitive	移动侦测灵敏度, 0 - 5, 越高越灵敏, 0xff 关闭
byEnableHandleMotion	是否处理移动侦测 0-否 1-是
byEnabledDisplay	启用移动侦测高亮显示, 0-否, 1-是
byRes	保留
struMotionHandleType	处理方式
struAlarmTime	布防时间
byRelRecordChan	告警触发的录象通道

```
typedef struct
```

```
{
    BYTE          byMotionScope[64][96];
    BYTE          byMotionSensitive;
    BYTE          byEnableHandleMotion;
    BYTE          byEnabledDisplay;
    BYTE          byRes;
    SRAI_HANDLEALARM struMotionHandleType;
```

```

        SRAI_SCHEDULETIME          struAlarmTime[MAX_DAYS][8];
        BYTE                        byRelRecordChan[MAX_CHANNUM];
    }SRAI_VIDEO_MOTION, *LPSRAI_VIDEO_MOTION;

```

B. 4. 8 图像通道参数见表 B. 30。

表 B. 30 图像通道参数

字段名称	含义
wChannel	视频通道号
byRes	保留
dwVideoFormat	视频制式 1-NTSC 2-PAL
dwShowChanName	预览的图象上是否显示通道名称, 0-不显示, 1-显示
wShowNameTopLeftX	通道名称显示位置的 x 坐标
wShowNameTopLeftY	通道名称显示位置的 y 坐标
struVILost	视频信号丢失告警
struMotion	移动侦测
struHideAlarm	遮挡告警
dwEnableHide	是否启动遮盖, 0-否, 1-是
struShelter	遮盖区域大小
dwShowOsd	预览的图象上是否显示 OSD, 0-不显示, 1-显示
wOSDTopLeftX	OSD 的 x 坐标
wOSDTopLeftY	OSD 的 y 坐标 0x01: XX-XX-XXXX 月日年 0x02: XXXX 年 XX 月 XX 日
byDispWeek	是否显示星期
byHourOSDType	OSD 小时制: 0-24 小时制, 1-12 小时制
byFontSize	字体大小, 0-16*16(中)/8*16(英), 1-32*32(中)/16*32(英), 2-64*64(中)/32*64(英)
byOSDType	0x03: XX 月 XX 日 XXXX 年 0x04: XX-XX-XXXX 日月年 0x05: XX 日 XX 月 XXXX 年 0x06: xx/xx/xxxx(月/日/年) 0x07: xxxx/xx/xx(年/月/日) 0x08: xx/xx/xxxx(日/月/年)
byOSDAtrib	OSD 属性: 透明, 闪烁 0x01: 透明, 闪烁 0x02: 透明, 不闪烁 0x03: 闪烁, 不透明 0x04: 不透明, 不闪烁
byRes1	保留

```
typedef struct
```

```
{
```

```
    WORD                wChannel;
```

```

BYTE                byRes[2];
DWORD               dwVideoFormat;
DWORD               dwShowChanName;
WORD                wShowNameTopLeftX;
WORD                wShowNameTopLeftY;
SRAI_VIDEO_LOST    struVILost;
SRAI_VIDEO_MOTION  struMotion;
SRAI_VIDEO_HIDE    struHideAlarm;
DWORD               dwEnableHide;
SRAI_SHELTER        struShelter[MAX_SHELTERNUM];
DWORD               dwShowOsd;
WORD                wOSDTopLeftX;
WORD                wOSDTopLeftY;
BYTE                byOSDType;
BYTE                byDispWeek;
BYTE                byOSDAAttrib;
BYTE                byHourOSDType;
BYTE                byFontSize;
BYTE                byRes1[3];
}SRAI_PICCFG, *LPSRAI_PICCFG;

```

B. 4. 9 码流压缩参数见表 B. 31。

表 B. 31 码流压缩参数

字段名称	含义
wChannel	视频通道号
byStreamNO	码流编号，0-主码流，1-子码流，2-子码流... 以此类推
byStreamType	码流类型 0-视频流，1-复合流
wWidth	图形宽度
wHeight	图形高度
byBitrateType	码率类型 0:变码率，1:定码率
byPicQuality	图象质量 0-最好 1-次好 2-较好 3-一般 4-较差 5-差
byRes	保留
dwVideoBitrate	码率，单位 Kbit，最小值 16
dwVideoFrameRate	帧率，0 表示全帧率
wIntervalFrameI	I 帧间隔
byRes1	保留
byIntervalBPFrame	0-BBP 帧；1-BP 帧；2-单 P 帧
wVideoEncType	视频编码类型
wAudioEncType	音频编码类型
byVideoEncComplexity	视频编码复杂度，0-低，1-中，2 高
byRes2	保留

```

typedef struct
{
    WORD        wChannel;
    BYTE        byStreamNO;
    BYTE        byStreamType;
    WORD        wWidth;
    WORD        wHeight;
    BYTE        byBitrateType;
    BYTE        byPicQuality;
    BYTE        byRes[2];
    DWORD       dwVideoBitrate;
    DWORD       dwVideoFrameRate;
    WORD        wIntervalFrameI;
    BYTE        byIntervalBPFrame;
    BYTE        byRes1;
    WORD        wVideoEncType;
    WORD        wAudioEncType;
    BYTE        byVideoEncComplexity;
    BYTE        byRes2[3];
}SRAI_COMPRESSIONCFG, *LPSRAI_COMPRESSIONCFG;

```

B. 4. 10 全天录像参数见表 B. 32。

表 B. 32 全天录像参数

字段名称	含义
wAllDayRecord	是否全天录像 0-否 1-是
byRecordType	录像类型 0: 定时录像 1: 移动侦测 2: 告警录像 3: 移动侦测   告警 4: 移动侦测&告警 5: 手动录像 6: 智能录像 其他值扩展
byRes	保留

```

typedef struct
{
    WORD        wAllDayRecord;
    BYTE        byRecordType;
    BYTE        byRes;
}SRAI_RECORDDAY, *LPSRAI_RECORDDAY;

```



B. 4. 11 时间段录像参数见表 B. 33。

表 B. 33 时间段录像参数

字段名称	含义
struRecordTime	时间段
byRecordType	录像类型 0:定时录像 1:移动侦测 2:告警录像 3:移动侦测 告警 4:移动侦测&告警 5:手动录像 6:智能录像 其他值扩展
byRes	保留

```
typedef struct
{
    SRAI_SCHEDULETIME    struRecordTime;
    BYTE                  byRecordType;
    BYTE                  byRes[3];
}SRAI_RECORDSCHED, *LPSRAI_RECORDSCHED;
```

B. 4. 12 通道录像参数见表 B. 34。

表 B. 34 通道录像参数

字段名称	含义
wChannel	视频通道号
byRes	保留
dwRecord	是否录像 0-否 1-是
struRecAllDay	全天录像
struRecordSched	时间段录像
dwRecordTime	录像延时长度 0-5 秒， 1-10 秒， 2-30 秒， 3-1 分钟， 4-2 分钟， 5-5 分钟， 6-10 分钟
dwPreRecordTime	预录时间 0-不预录 1-5 秒 2-10 秒 3-15 秒 4-20 秒 5-25 秒 6-30 秒 7-0xffffffff(尽可能预录)
dwRecorderDuration	录像保存的最长时间
byRedundancyRec	是否冗余录像,重要数据双备份: 0 否, 1 是
byAudioRec	录像时复合流编码时是否记录音频数据;
byRes1	保留

```
typedef struct
{
```

```

WORD          wChannel;
BYTE          byRes[2];
DWORD         dwRecord;
SRAI_RECORDDAY struRecAllDay[MAX_DAYS];
SRAI_RECORDSCHED struRecordSched[MAX_DAYS][8];
DWORD         dwRecordTime;
DWORD         dwPreRecordTime;
DWORD         dwRecorderDuration;
BYTE          byRedundancyRec;
BYTE          byAudioRec;
BYTE          byRes1[2];
}SRAI_RECORDCFG, *LPSRAI_RECORDCFG;

```

B. 4. 13 RS232 串口参数见表 B. 35。

表 B. 35 RS232 串口参数

字段名称	含义
wChannel	串口通道号
byRes	保留
dwBaudRate	波特率，单位 bps
byDataBit	数据位数
byStopBit	停止位数
byParity	校验 0—无校验，1—奇校验，2—偶校验
byFlowcontrol	0—无，1—软流控，2—硬流控
dwWorkMode	工作模式，0—232 串口用于 PPP 拨号，1—232 串口用于参数控制，2—透明通道，其他扩展

```

typedef struct
{
    WORD          wChannel;
    BYTE          byRes[2];
    DWORD         dwBaudRate;
    BYTE          byDataBit;
    BYTE          byStopBit;
    BYTE          byParity;
    BYTE          byFlowcontrol;
    DWORD         dwWorkMode;
} SRAI_RS232CFG, *LPSRAI_RS232CFG;

```

B. 4. 14 告警输入参数见表 B. 36。

表 B. 36 告警输入参数

字段名称	含义
sAlarmInName	告警输入名称
byAlarmType	告警器类型, 0: 常开, 1: 常闭
byAlarmInHandle	是否处理 0-不处理 1-处理
wChannel	告警输入触发智能识别通道
struAlarmHandleType	处理方式
struAlarmTime	布防时间
byRelRecordChan	告警触发的录像通道, 为 1 表示触发该通道
byEnablePreset	是否调用预置点 0-否, 1-是
byPresetNo	调用的云台预置点序号, 一个告警输入可以调用多个通道的云台预置点, 0xff 表示不调用预置点。
byEnableCruise	是否调用巡航 0-否, 1-是
byCruiseNo	巡航
byEnablePtzTrack	是否调用轨迹 0-否, 1-是
byPTZTrack	调用的云台的轨迹序号

```
typedef struct
```

```
{
    CHAR                sAlarmInName[NAME_LEN];
    BYTE                byAlarmType;
    BYTE                byAlarmInHandle;
    WORD                wChannel;
    SRAI_HANDLEALARM    struAlarmHandleType;
    SRAI_SCHEDTIME      struAlarmTime[MAX_DAYS][8];
    BYTE                byRelRecordChan[MAX_CHANNUM];
    BYTE                byEnablePreset[MAX_CHANNUM];
    BYTE                byPresetNo[MAX_CHANNUM];
    BYTE                byEnableCruise[MAX_CHANNUM];
    BYTE                byCruiseNo[MAX_CHANNUM];
    BYTE                byEnablePtzTrack[MAX_CHANNUM];
    BYTE                byPTZTrack[MAX_CHANNUM];
}SRAI_ALARMINCFG, *LPSRAI_ALARMINCFG;
```

B. 4. 15 夏时令参数见表 B. 37。

表 B. 37 夏时令参数

字段名称	含义
dwEnableDST	是否启用夏时制 0-不启用 1-启用
byDSTBias	夏令时偏移值, 单位分钟
byRes	保留
struBeginPoint	夏时制开始时间

表 B. 37 (续)

字段名称	含义
struEndPoint	夏时制停止时间

```
typedef struct
{
    DWORD          dwEnableDST;
    BYTE           byDSTBias;
    BYTE           byRes[3];
    SRAI_TIMEPOINT struBeginPoint;
    SRAI_TIMEPOINT struEndPoint;
}SRAI_ZONEANDDST, *LPSRAI_ZONEANDDST;
```

B. 4. 16 单字符参数见表 B. 38。

表 B. 38 单字符参数

字段名称	含义
wShowString	预览的图象上是否显示字符, 0-不显示, 1-显示
wStringSize	该行字符的长度, 不能大于 44 个字符
wShowStringTopLeftX	字符显示位置的 x 坐标
wShowStringTopLeftY	字符名称显示位置的 y 坐标
sString	显示的字符内容

```
typedef struct
{
    WORD          wShowString;
    WORD          wStringSize;
    WORD          wShowStringTopLeftX;
    WORD          wShowStringTopLeftY;
    CHAR          sString[44];
}SRAI_SHOWSTRINGINFO, *LPSRAI_SHOWSTRINGINFO;
```

B. 4. 17 字符叠加参数见表 B. 39。

表 B. 39 字符叠加参数

字段名称	含义
wChannel	视频通道号
byRes	保留
struStringInfo	要显示的字符内容

```
typedef struct
{
    WORD          wChannel;
```

```

        BYTE                byRes[2];
        SRAI_SHOWSTRINGINFO struStringInfo[MAX_STRINGNUM];
    }SRAI_SHOWSTRING, *LPSRAI_SHOWSTRING;

```

## B.5 接口调用参数

B.5.1 登录信息见表 B.40。

表 B.40 登录信息

字段名称	含义
IP	监控资源地址
wPort	监控资源端口
byRes	保留
sName	用户名
sPassword	密码
wReconnectTime	断线重连时间，单位秒
byRes1	保留

```

typedef struct {
    SRAI_IPADDR    IP;
    WORD           wPort;
    BYTE           byRes[2];
    CHAR           sName[NAME_LEN];
    CHAR           sPassword[PASSWD_LEN];
    WORD           wReconnectTime;
    BYTE           byRes1[2];
}SRAI_LOGIN_INFO, *LPSRAI_LOGIN_INFO;

```

B.5.2 注册服务信息见表 B.41。

表 B.41 注册服务信息

字段名称	含义
IP	本地网络地址
wPort	本地端口
byRes	保留

```

typedef struct {
    SRAI_IPADDR    IP;
    WORD           wPort;
    BYTE           byRes[2];
}SRAI_START_SERVER, *LPSRAI_START_SERVER;

```

B. 5. 3 获取或设置参数信息见表 B. 42。

表 B. 42 获取或设置参数信息

字段名称	含义
dwType	参数类型
pBuf	参数数据
dwBufLen	数据长度

```
typedef struct
{
    DWORD    dwType;
    DWORD    dwBufLen;
    BYTE *   pBuf;
}SRAI_PARAM, *LPSRAI_PARAM;
```

B. 5. 4 开启实时预览见表 B. 43。

表 B. 43 开启实时预览

字段名称	含义
wChannel	通道号
byStreamNO	播放的码流编号，0-主码流，1-子码流，2-子码流... 以此类推
byLinkMode	连接方式 0-TCP，1-UDP
DataCBK	数据回调函数
pUser	用户数据，回调函数中带回

```
typedef struct
{
    WORD        wChannel;
    BYTE        byStreamNO;
    BYTE        byLinkMode;
    fDataCallBack    DataCBK;
    void*       pUser;
}SRAI_REALPLAY, *LPSRAI_REALPLAY;
```

B. 5. 5 回放或下载录像见表 B. 44。

表 B. 44 回放或下载录像

字段名称	含义
wChannel	通道号
byType	录像方式，0-时间方式，1-文件方式
byRes	保留
sName	文件方式，录像名称，当使用时间方式时该字段无效

表 B. 44 (续)

字段名称	含义
dwStartPos	文件方式，开始回放或下载的位置，相对于该录像文件开始时间的偏移值，取值范围为 0 至该录像文件总长度，单位为秒，当使用时间方式时该字段无效
StartTime	时间方式，开始时间，当使用文件方式时该字段无效
EndTime	时间方式，结束时间，当使用文件方式时该字段无效
DataCBK	数据回调函数
pUser	用户数据，回调函数中带回

```
typedef struct
```

```
{
    WORD        wChannel;
    BYTE        byType;
    BYTE        byRes;
    BYTE        sName[NAME_LEN];
    DWORD       dwStartPos;
    SRAI_TIME   StartTime;
    SRAI_TIME   EndTime;
    fDataCallBack DataCBK;
    void*       pUser;
}SRAI_PLAYBACK, *LPSRAI_PLAYBACK;
```

B. 5. 6 回放录像控制见表 B. 45。

表 B. 45 回放录像控制

字段名称	含义
dwCmd	回放录像控制命令
pInBuffer	指向输入参数的指针
dwInLen	输入参数的长度
pOutBuffer	指向输出参数的指针
*lpOutLen	输出参数的长度

```
typedef struct
```

```
{
    DWORD       dwCmd;
    BYTE *      pInBuffer;
    DWORD       dwInLen;
    BYTE *      pOutBuffer;
    DWORD       dwOutLen;
}SRAI_PLAYBACK_CONTROL, *LPSRAI_PLAYBACK_CONTROL;
```

B. 5. 7 抓拍见表 B. 46。

表 B. 46 抓拍

字段名称	含义
wChannel	通道号
byType	类型，0-远程抓拍，1-本地抓拍至文件，2-本地抓拍至内存
byRes	保留
sName	图片文件全路径包括名称，本地抓拍至文件时有效
pBuf	图片数据，本地抓拍至内存时有效
dwBufLen	图片数据长度，本地抓拍至内存时有效

typedef struct

```
{
    WORD        wChannel;
    BYTE        byType;
    BYTE        byRes;
    CHAR        sName[NAME_LEN];
    DWORD       dwBufLen;
    BYTE *      pBuf;
}SRAI_SNAPSHOT_PARAM, *LPSRAI_SNAPSHOT_PARAM;
```

B. 5. 8 云台控制见表 B. 47。

表 B. 47 云台控制

字段名称	含义
wCommand	控制类型
byStop	停止标志
byRes	保留
dwSpeed	云台速度
dwPresetNO	预置点号，设置和删除预置点时有效，范围 1-255
byCruiseRoute	巡航路径，最多支持 32 条路径（序号从 1 开始），巡航操作时有效
byCruisePoint	巡航点，最多支持 32 个点（序号从 1 开始），巡航操作时有效
dwTime	巡航点停顿时间，巡航操作时有效
byRes1	保留

typedef struct

```
{
    WORD        wChannel;
    WORD        wCommand;
    BYTE        byStop;
    BYTE        byRes[3];
    DWORD       dwSpeed;
    DWORD       dwPresetNO;
    BYTE        byCruiseRoute;
```



```

    BYTE        byCruisePoint;
    BYTE        byRes1[2];
    DWORD       dwTime;
}SRAI_PTZ_CTRL, *LPSRAI_PTZ_CTRL;

```

B. 5. 9 手动录像见表 B. 48。

表 B. 48 手动录像

字段名称	含义
wChannel	视频通道
byRes	保留

```

typedef struct
{
    WORD        wChannel;
    BYTE        byRes[2];
}SRAI_START_RECORD, *LPSRAI_START_RECORD;

```

B. 5. 10 查询图片或录像见表 B. 49。

表 B. 49 查询图片或录像

字段名称	含义
wChannel	视频通道
byRes	保留
StartTime	开始时间
EndTime	结束时间

```

typedef struct
{
    WORD        wChannel;
    BYTE        byRes[2];
    SRAI_TIME    StartTime;
    SRAI_TIME    EndTime;
}SRAI_QUERY, *LPSRAI_QUERY;

```

B. 5. 11 语音对讲见表 B. 50。

表 B. 50 语音对讲

字段名称	含义
wChannel	视频通道
byRes	保留
DataCBK	数据回调函数
pUser	用户数据，回调函数中带回

```
typedef struct
{
    WORD          wChannel;
    BYTE          byRes[2];
    fDataCallBack DataCBK;
    void*         pUser;
} SRAI_TALK, * LPSRAI_TALK;
```

B. 5. 12 日志见表 B. 51。

表 B. 51 日志

字段名称	含义
LogTime	日志时间
dwType	日志类型
sPanelUser	操作面板的用户名
sNetUser	网络操作的用户名
sRemoteHostAddr	远程主机地址
dwChannel	通道号
dwDisNumber	硬盘号
字段名称	含义
dwAlarmInPort	告警输入端口
dwAlarmOutPort	告警输出端口

```
typedef struct
{
    SRAI_TIME      LogTime;
    DWORD          dwType;
    CHAR           sPanelUser[NAME_LEN];
    CHAR           sNetUser[NAME_LEN];
    CHAR           sRemoteHostAddr[16];
    DWORD          dwChannel;
    DWORD          dwDiskNumber;
    DWORD          dwAlarmInPort;
    DWORD          dwAlarmOutPort;
} SRAI_LOG, * LPSRAI_LOG;
```

## 附录 C (规范性附录) 常量定义

### C.1 布尔值

布尔值见表C.1。

表 C.1 布尔值

描述	预定义值	说明
TRUE	1	布尔值为真
FALSE	0	布尔值为假

### C.2 一般常量

一般常量见表C.2。

表 C.2 一般常量

描述	预定义值	说明
MAX_DAYS	0x00000007	每周天数
PHONENUMBER_LEN	0x00000020	号码长度
NAME_LEN	0x00000040	名称长度
SERIES_LEN	0x00000080	序列号长度
ERROR_MSG_LEN	0x00000100	错误信息长度
PASSWD_LEN	0x00000020	密码长度
SERIALNO_LEN	0x00000040	序列号长度
MACADDR_LEN	0x00000006	MAC 地址长度
MAX_ETHERNET	0x00000002	监控资源可配置以太网数
MAX_DOMAIN_NAME	0x00000040	最大域名长度
MAX_NETWORK_CARD	0x00000004	监控资源可配最大网卡数目
MAX_ANALOG_ALARMOUT	0x00000020	最大模拟告警输出路数
MAX_IP_ALARMOUT	0x00000040	允许加入的最多告警输出路数
MAX_ALARMOUT	0x00000060	最大告警输出路数 (MAX_ANALOG_ALARMOUT+ MAX_IP_ALARMOUT)
MAX_ANALOG_CHANNUM	0x00000020	最大模拟通道数
MAX_IP_CHANNEL	0x00000020	允许加入的最多 IP 通道数
MAX_CHANNUM	0x00000040	最大通道数 (MAX_ANALOG_CHANNUM+ MAX_IP_CHANNEL)
MAX_SHELTERNUM	0x00000004	最大遮挡区域数
MAX_SERIAL_PORT	0x00000008	最大 232 串口数
MAX_STRINGNUM	0x00000008	最大 OSD 字符行数

### C.3 错误代码

错误代码见表C. 3。

表 C. 3 错误代码

描述	预定义值	说明
SRAI_ERROR_NOERROR	0x00000000	没有错误
SRAI_ERROR_NOINIT	0x00000001	未初始化
SRAI_ERROR_USERID	0x00000002	错误的用户 ID
SRAI_ERROR_INPUT_PARAM	0x00000003	输入参数错误
SRAI_ERROR_TIMEOUT	0x00000004	请求超时
SRAI_ERROR_UNSUPPORT	0x00000005	接口不支持
SRAI_ERROR_ORDER	0x00000006	接口调用顺序错误
SRAI_ERROR_FILE_CREATE	0x00000007	文件无法创建
SRAI_ERROR_RESOURCE_NOTEXIST	0x00000008	监控资源不存在
SRAI_ERROR_CHAN_NOTEXIST	0x00000009	通道不存在
SRAI_ERROR_PIC_NOTEXIST	0x0000000A	图片不存在
SRAI_ERROR_REC_NOTEXIST	0x0000000B	录像不存在
SRAI_ERROR_DISCONNECT	0x0000000C	连接已断开
SRAI_ERROR_SYSTEM_MEM	0x0000000D	系统内存不足
SRAI_ERROR_DISK_FULL	0x0000000E	硬盘空间不足
SRAI_ERROR_DISK_WRITE	0x0000000F	硬盘写失败
SRAI_ERROR_DISK_READ	0x00000010	硬盘读失败
SRAI_ERROR_FILE_BRK	0x00000011	文件损坏

C. 4 行业类型

行业类型见表C. 4。

表 C. 4 行业类型

描述	预定义值	说明
SRAI_IND_GL	0x00	公路
SRAI_IND_HD	0x01	航道
SRAI_IND_YS	0x02	运管
SRAI_IND_ZJ	0x03	质监
SRAI_IND_GA	0x04	公安
SRAI_IND_JC	0x05	机场
SRAI_IND_HK	0x06	航空
SRAI_IND_TL	0x07	铁路
SRAI_IND_DT	0x08	地铁
SRAI_IND_CZ	0x09	场站
SRAI_IND_GD	0x0A	工地
SRAI_IND_HS	0x0B	海事管理

## C.5 消息类型

消息类型见表C.5。

表 C.5 消息类型

描述	预定义值	消息回调函数中 pBuf 对应的数据结构	说明
SRAI_MSG_RESOURCE_ONLINE	0x00000000	LPSRAI_RESOURCE_INFO	监控资源连接成功
SRAI_MSG_RESOURCE_OFFLINE	0x00000001	LPSRAI_RESOURCE_INFO	监控资源断开连接
SRAI_MSG_USER	0x00001000-0x0000FFFF	未定义	扩展消息类型

## C.6 操作类型

操作类型见表C.6。

表 C.6 操作类型

描述	预定义值	操作回调函数中 pBuf 对应的数据结构	说明
SRAI_OPT_LOGIN_OK	0x00000000	无效	登录成功
SRAI_OPT_LOGIN_FAIL	0x00000001	LPSRAI_ERROR_INFO	登录失败
SRAI_OPT_REALPLAY_OK	0x00000002	无效	实时播放成功
SRAI_OPT_REALPLAY_FAIL	0x00000003	LPSRAI_ERROR_INFO	实时播放失败
SRAI_OPT_QUERY_PIC_OK	0x00000004	LPSRAI_PICTURE_GROUP	查询图片成功
SRAI_OPT_QUERY_PIC_FAIL	0x00000005	LPSRAI_ERROR_INFO	查询图片失败
SRAI_OPT_QUERY_REC_OK	0x00000006	LPSRAI_RECORD_GROUP	录像查询成功
SRAI_OPT_QUERY_REC_FAIL	0x00000007	LPSRAI_ERROR_INFO	录像查询失败
SRAI_OPT_DOWNLOAD_PIC_OK	0x00000008	无效	下载图片成功
SRAI_OPT_DOWNLOAD_PIC_FAIL	0x00000009	LPSRAI_ERROR_INFO	下载图片失败
SRAI_OPT_DOWNLOAD_REC_OK	0x0000000A	无效	回放录像成功
SRAI_OPT_DOWNLOAD_REC_FAIL	0x0000000B	LPSRAI_ERROR_INFO	回放录像失败
SRAI_OPT_PLAYBACK_OK	0x0000000C	无效	回放成功
SRAI_OPT_PLAYBACK_FAIL	0x0000000D	LPSRAI_ERROR_INFO	回放失败

## C.7 通道类型

通道类型见表C.7。

表 C.7 通道类型

描述	预定义值	说明
SRAI_CHAN_VIDEO	0x00	视频通道

表C.7（续）

描述	预定义值	说明
SRAI_CHAN_ALARMIN	0x01	告警输入通道
SRAI_CHAN_ALARMOUT	0x02	告警输出通道
SRAI_CHAN_DECODE	0x03	解码通道

## C.8 参数类型

参数类型见表C.8。

表 C.8 参数类型

描述	预定义值	LPSRAI_PARAM 中参数数据对应的数据结构	说明
SRAI_CMD_RESOURCECFG	0x00000000	LPSRAI_RESOURCE_INFO	监控资源参数
SRAI_CMD_NETCFG	0x00000001	LPSRAI_NETCFG	网络参数
SRAI_CMD_PICCFG	0x00000002	LPSRAI_PICCFG	图象参数
SRAI_CMD_COMPRESSCFG	0x00000003	LPSRAI_COMPRESSIONCFG	压缩参数
SRAI_CMD_RECORDCFG	0x00000004	LPSRAI_RECORDCFG	通道录像参数
SRAI_CMD_RS232CFG	0x00000005	LPSRAI_RS232CFG	232 串口参数
SRAI_CMD_ALARMINCFG	0x00000006	LPSRAI_ALARMINCFG	告警输入参数
SRAI_CMD_TIMECFG	0x00000007	LPSRAI_TIME	监控资源时间
SRAI_CMD_ZONEANDDST	0x00000008	LPSRAI_ZONEANDDST	时区和夏时制参数
SRAI_CMD_SHOWSTRING	0x00000009	LPSRAI_SHOWSTRING	叠加字符参数

## C.9 告警类型

告警类型见表C.9。

表 C.9 告警类型

描述	预定义值	pAlarmInfo 对应结构体	LPSRAI_ALARM_INFO 中 wChannel 说明	说明
SRAI_ALARM_CPU_HIGH	0x00000000	LPSRAI_ALARM_INFO	无效	CPU 过高
SRAI_ALARM_TEMPERATURE_ABNORMAL	0x00000001	LPSRAI_ALARM_INFO	无效	温度异常
SRAI_ALARM_CLOCK_ABNORMAL	0x00000002	LPSRAI_ALARM_INFO	无效	时钟异常
SRAI_ALARM_FAN_ABNORMAL	0x00000003	LPSRAI_ALARM_INFO	无效	风扇状态异常
SRAI_ALARM_MEMORY_ABNORMAL	0x00000004	LPSRAI_ALARM_INFO	无效	内存状态异常
SRAI_ALARM_DISK	0x0000000A	LPSRAI_ALARM_INFO	硬盘号	硬盘告警
SRAI_ALARM_DISK_FULL	0x0000000B	LPSRAI_ALARM_INFO	硬盘号	硬盘满
SRAI_ALARM_DISK_WRError	0x0000000C	LPSRAI_ALARM_INFO	硬盘号	硬盘读写出错

表 C.9 (续)

描述	预定义值	pAlarmInfo 对应结构体	LPSRAI_ALARM_INFO 中 wChannel 说明	说明
SRAI_ALARM_VIDEO_LOST	0x0000000D	LPSRAI_ALARM_INFO	视频通道号	视频丢失
SRAI_ALARM_VIDEO_ABNORMAL	0x0000000E	LPSRAI_ALARM_INFO	视频通道号	图形异常
SRAI_ALARM_VIDEO_MISMATCH	0x0000000F	LPSRAI_ALARM_INFO	视频通道号	制式不匹配
SRAI_ALARM_VIDEO_MOTION	0x00000010	LPSRAI_ALARM_INFO	视频通道号	移动侦测
SRAI_ALARM_VIDEO_COVERED	0x00000011	LPSRAI_ALARM_INFO	视频通道号	视频遮挡
SRAI_ALARM_INPUT	0x00000012	LPSRAI_ALARM_INFO	视频通道号	告警输入
SRAI_ALARM_ACCESS_OPEN	0x00000013	LPSRAI_ALARM_INFO	告警输入通道号	门禁开
SRAI_ALARM_ACCESS_CLOSE	0x00000014	LPSRAI_ALARM_INFO	告警输入通道号	门禁关
SRAI_ALARM_INFRARED	0x00000015	LPSRAI_ALARM_INFO	告警输入通道号	红外告警
SRAI_ALARM_OUTPUT	0x00000016	LPSRAI_ALARM_INFO	告警输入通道号	告警输入参数
SRAI_ALARM_NET	0x00000017	LPSRAI_ALARM_INFO	无效	网络异常
SRAI_ALARM_SRAI_BROKEN	0x00000018	LPSRAI_ALARM_INFO	无效	网络断开
SRAI_ALARM_IP_CONFLICT	0x00000019	LPSRAI_ALARM_INFO	无效	IP 地址冲突
SRAI_ALARM_GPRS	0x0000002A	LPSRAI_ALARM_INFO	无效	GPRS 通信异常
SRAI_ALARM_3G	0x0000002B	LPSRAI_ALARM_INFO	无效	3G 通信异常
SRAI_ALARM_SIM_CARD	0x0000002C	LPSRAI_ALARM_INFO	无效	SIM 卡异常

## C.10 视频编码类型

视频编码类型见表C.10。

表 C.10 视频编码类型

描述	预定义值	说明
VIDEO_NULL	0x0000	无视频
VIDEO_H264	0x0001	H.264, 符合 ISO/IEC 14496.10 (MPEG-4 AVC part 10; H.264) 标准的规定

## C.11 音频编码类型

音频编码类型见表C.11。

表 C.11 音频编码类型

描述	预定义值	说明
AUDIO_NULL	0x0000	无音频

表 C. 11 (续)

描述	预定义值	说明
AUDIO_AAC	0x0001	AAC 编码

## C. 12 帧类型

帧类型见表C. 12。

表 C. 12 帧类型

描述	预定义值	说明
SRAI_FRAME_AUDIO_PCM	0x0000	音频帧, PCM
SRAI_FRAME_VIDEO_I420	0x0001	视频帧, I420 排列顺序" Y0-Y1-Y2..." ; " V0-V1..." ; " U0-U1..."
SRAI_FRAME_VIDEO_YV12	0x0002	视频帧, YV12 排列顺序" Y0-Y1-Y2..." ; " U0-U1..." ; " V0-V1..."
SRAI_FRAME_VIDEO_NV12	0x0003	视频帧, NV12 排列顺序" Y0-Y1-Y2..." ; " U0-V0-U1-V1..."

## C. 13 数据类型

0x0000-0x00FF 为保留区, 0x0100-0x0FFF 为扩展区。数据类型见表 C. 13。

表 C. 13 数据类型

描述	预定义值	说明
SRAI_DATA_VIDEO	0x0000	视频数据
SRAI_DATA_AUDIO	0x0001	音频数据

## C. 14 云台控制类型

0x0000-0x00FF 为保留区, 0x0100-0x0FFF 为扩展区。云台控制类型见表 C. 14。

表 C. 14 云台控制类型

描述	预定义值	SRAI_PTZ_CTRL 中 dwSpeed 值	说明
SRAI_PTZ_LIGHT_PWRON	0x0002	无效	接通灯光电源
SRAI_PTZ_WIPER_PWRON	0x0003	无效	接通雨刷开关
SRAI_PTZ_FAN_PWRON	0x0004	无效	接通风扇开关
SRAI_PTZ_HEATER_PWRON	0x0005	无效	接通加热器开关
SRAI_PTZ_AUX_PWRON1	0x0006	无效	接通辅助监控资源开关
SRAI_PTZ_AUX_PWRON2	0x0007	无效	接通辅助监控资源开关
SRAI_PTZ_SET_PRESET	0x0008	无效	设置预置点



表C. 14（续）

描述	预定义值	SRAI_PTZ_CTRL 中 dwSpeed 值	说明
SRAI_PTZ_CLE_PRESET	0x0009	无效	清除预置点
SRAI_PTZ_ZOOM_IN	0x000A	有效	焦距变大
SRAI_PTZ_ZOOM_OUT	0x000B	有效	焦距变小
SRAI_PTZ_FOCUS_NEAR	0x000C	有效	焦点前调
SRAI_PTZ_FOCUS_FAR	0x000D	有效	焦点后调
SRAI_PTZ_IRIS_OPEN	0x000E	有效	光圈扩大
SRAI_PTZ_IRIS_CLOSE	0x000F	有效	光圈缩小
SRAI_PTZ_UP	0x0010	有效	上
SRAI_PTZ_DOWN	0x0011	有效	下
SRAI_PTZ_LEFT	0x0012	有效	左
SRAI_PTZ_RIGHT	0x0013	有效	右
SRAI_PTZ_UP_LEFT	0x0014	有效	左上
SRAI_PTZ_UP_RIGHT	0x0015	有效	右上
SRAI_PTZ_DOWN_LEFT	0x0016	有效	左下
SRAI_PTZ_DOWN_RIGHT	0x0017	有效	右下
SRAI_PTZ_AUTO	0x0018	有效	自动扫描
SRAI_PTZ_FILL_PRE_SEQ	0x0019	无效	将预置点加入巡航序列
SRAI_PTZ_SET_SEQ_DWELL	0x001A	无效	设置巡航点停顿时间
SRAI_PTZ_SET_SEQ_SPEED	0x001B	无效	设置巡航速度
SRAI_PTZ_CLE_PRE_SEQ	0x001C	无效	将预置点从巡航序列中删除
SRAI_PTZ_STA_MEM_CRUISE	0x001D	无效	开始记录轨迹
SRAI_PTZ_STO_MEM_CRUISE	0x001E	无效	停止记录轨迹
SRAI_PTZ_RUN_CRUISE	0x001F	无效	开始轨迹
SRAI_PTZ_RUN_SEQ	0x0020	无效	开始巡航
SRAI_PTZ_STOP_SEQ	0x0021	无效	停止巡航
SRAI_PTZ_GOTO_PRESET	0x0022	无效	快球转到预置点
SRAI_PTZ_DEL_SEQ	0x0023	无效	删除巡航路径
SRAI_PTZ_PAN_CIRCLE	0x0024	有效	云台自动圆周扫描

## C. 15 回放录像控制命令

回放录像控制命令见表C. 15。

表 C.15 回放录像控制命令

描述	预定义值	说明	SRAI_PLAYBACK 中 pInBuffer	SRAI_PLAYBACK 中 pOutBuffer
SRAI_PLAYBACK_START	0x0001	开始播放	一个 4 字节整形的偏移量，若该值为 0 表示从文件的起始位置播放，若该值不为 0 则表示断点续传的文件位置	无效
SRAI_PLAYBACK_PAUSE	0x0002	暂停播放	无效	无效
SRAI_PLAYBACK_RESTART	0x0003	恢复播放	无效	无效
SRAI_PLAYBACK_FAST	0x0004	快放	无效	无效
SRAI_PLAYBACK_SLOW	0x0005	慢放	无效	无效
SRAI_PLAYBACK_NORMAL	0x0006	正常速度播放，在暂停后调用将恢复暂停前的速度播放	无效	无效
SRAI_PLAYBACK_FRAME	0x0007	单帧播放，恢复正常播放使用 SRAI_PLAYBACK_PLAYNORMAL 命令	无效	无效
SRAI_PLAYBACK_STARTAUDIO	0x0008	打开声音	无效	无效
SRAI_PLAYBACK_STOPAUDIO	0x0009	关闭声音	无效	无效
SRAI_PLAYBACK_AUDIOVOLUME	0x000A	调节音量	一个 4 字节整型的数值，取值范围 0x0000-0xFFFF	无效
SRAI_PLAYBACK_SETPOS	0x000B	改变回放的进度	一个 4 字节整型的进度值，范围 0-100	无效
SRAI_PLAYBACK_GETPOS	0x000C	获取回放的进度	无效	一个 4 字节整型的进度值，范围 0-100
SRAI_PLAYBACK_GETTIME	0x000D	获取当前已经播放的时间，按文件回放时有效	无效	一个 4 字节整型的时间值，单位秒
SRAI_PLAYBACK_GETFRAME	0x000E	获取当前已经播放的帧数，按文件回放时有效	无效	一个 4 字节整型的帧数值

表C. 15（续）

描述	预定义值	说明	SRAI_PLAYBACK 中 pInBuffer	SRAI_PLAYBACK 中 pOutBuffer
SRAI_PLAYBACK_GETTOTALFRAMES	0x000F	获取当前播放文件总的帧数，按文件回放时有效	无效	一个 4 字节整型的帧数值
SRAI_PLAYBACK_GETTOTALTIME	0x0010	获取当前播放文件总的时间，按文件回放时有效	无效	一个 4 字节整型的时间值，单位秒
SRAI_PLAYBACK_SETSPEED	0x0011	设置回放速度	一个 4 字节整型的速度值,0 表示最大速度播放且不抽帧，正数为快放，负数为慢放，如： 4 表示 4 倍速播放 -4 表示 1/4 倍速播放	无效
SRAI_PLAYBACK_SETTIME	0x0012	按绝对时间定位	LPSRAI_TIME	无效
SRAI_PLAYBACK_FORWARD	0x0013	倒放切换为正放	无效	无效
SRAI_PLAYBACK_REVERSE	0x0014	正放切换为倒放	无效	无效

## C. 16 能力集类型

能力集类型见表C. 16。

表 C. 16 能力集类型

描述	预定义值	说明
SRAI_CAPABILITY_BASICINFO	0x0001	设备基本信息
SRAI_CAPABILITY_HSW	0x0002	设备软硬件
SRAI_CAPABILITY_PTZ	0x0003	云台控制
SRAI_CAPABILITY_CONFIG	0x0004	配置能力集合

附录 D  
(规范性附录)  
头文件定义

```
#ifndef _SRAI_H_
#define _SRAI_H_

#ifdef SRAI_EXPORTS
#define SRAI_API extern "C" __declspec(dllexport)
#else
#define SRAI_API extern "C" __declspec(dllimport)
#endif

#ifdef WIN32
#define STDCALL __stdcall
#else
#ifdef i386
#define STDCALL __attribute__((stdcall))
#else
#define STDCALL
#endif
#endif

typedef __int64          LONG;
typedef unsigned __int32  DWORD;
typedef unsigned __int16  WORD;
typedef unsigned char     BYTE ;

#ifndef TRUE
#define TRUE  1
#endif
#ifndef FALSE
#define FALSE 0
#endif
#ifndef NULL
#define NULL  0
#endif

// 一般常量
#define MAX_DAYS          0x00000007
#define PHONENUMBER_LEN   0x00000020
#define NAME_LEN           0x00000040
#define SERIES_LEN        0x00000080
#define ERROR_MSG_LEN     0x00000100
```

```

#define PASSWD_LEN          0x00000020
#define SERIALNO_LEN        0x00000040
#define MACADDR_LEN         0x00000006
#define MAX_ETHERNET        0x00000002
#define MAX_DOMAIN_NAME     0x00000040
#define MAX_NETWORK_CARD    0x00000004
#define MAX_ANALOG_ALARMOUT 0x00000020
#define MAX_IP_ALARMOUT     0x00000040
#define MAX_ALARMOUT        0x00000060
#define MAX_ANALOG_CHANNUM 0x00000020
#define MAX_IP_CHANNEL      0x00000020
#define MAX_CHANNUM         0x00000040
#define MAX_SHELTERNUM      0x00000004
#define MAX_SERIAL_PORT     0x00000008
#define MAX_STRINGNUM       0x00000008

```

// 错误代码

```
enum
```

```
{
```

```
    SRAI_ERROR_NOERROR = 0x00000000, //没有错误
```

```
    SRAI_ERROR_NOINIT, //未初始化
```

```
    SRAI_ERROR_USERID, //错误的用户 ID
```

```
    SRAI_ERROR_INPUT_PARAM, //输入参数错误
```

```
    SRAI_ERROR_TIMEOUT, //请求超时
```

```
    SRAI_ERROR_UNSUPPORT, //接口不支持
```

```
    SRAI_ERROR_ORDER, //接口调用顺序错误
```

```
    SRAI_ERROR_FILE_CREATE, //文件无法创建
```

```
    SRAI_ERROR_RESOURCE_NOTEXIST, //监控资源不存在
```

```
    SRAI_ERROR_CHAN_NOTEXIST, //通道不存在
```

```
    SRAI_ERROR_PIC_NOTEXIST, //图片不存在
```

```
    SRAI_ERROR_REC_NOTEXIST, //录像不存在
```

```
    SRAI_ERROR_DISCONNECT, //连接已断开
```

```
    SRAI_ERROR_SYSTEM_MEM, //系统内存不足
```

```
    SRAI_ERROR_DISK_FULL, //硬盘空间不足
```

```
    SRAI_ERROR_DISK_WRITE, //硬盘写失败
```

```
    SRAI_ERROR_DISK_READ, //硬盘读失败
```

```
    SRAI_ERROR_FILE_BRK, //文件损坏
```

```
};
```

// 行业类型

```
enum{
```

```
    SRAI_IND_GL = 0x00, //公路
```

```
    SRAI_IND_HD, //航道
```

```

    SRAI_IND_YS, //运管
    SRAI_IND_ZJ, //质监
    SRAI_IND_GA, //公安
    SRAI_IND_JC, //机场
    SRAI_IND_HK, //航空
    SRAI_IND_TL, //铁路
    SRAI_IND_DT, //地铁
    SRAI_IND_CZ, //场站
    SRAI_IND_GD, //工地
};

// 消息类型
enum
{
    SRAI_MSG_RESOURCE_ONLINE = 0x00000000, //监控资源连接成功
    SRAI_MSG_RESOURCE_OFFLINE = 0x00000001, //监控资源断开连接
    SRAI_MSG_USER = 0x00001000, //未定义    扩展消息类型
};

// 操作类型
enum
{
    SRAI_OPT_LOGIN_OK    = 0x00000000, //登陆成功
    SRAI_OPT_LOGIN_FAIL, //登陆失败
    SRAI_OPT_REALPLAY_OK, //实时播放成功
    SRAI_OPT_REALPLAY_FAIL, //实时播放失败
    SRAI_OPT_QUERY_PIC_OK, //查询图片成功
    SRAI_OPT_QUERY_PIC_FAIL, //查询图片失败
    SRAI_OPT_QUERY_REC_OK, //录像查询成功
    SRAI_OPT_QUERY_REC_FAIL, //录像查询失败
    SRAI_OPT_DOWNLOAD_PIC_OK, //下载图片成功
    SRAI_OPT_DOWNLOAD_PIC_FAIL, //下载图片失败
    SRAI_OPT_DOWNLOAD_REC_OK, //下载录像成功
    SRAI_OPT_DOWNLOAD_REC_FAIL, //下载录像失败
    SRAI_OPT_PLAYBACK_OK, //回放成功
    SRAI_OPT_PLAYBACK_FAIL, //回放失败
};

// 通道类型
enum
{
    SRAI_CHAN_VIDEO = 0x00, //视频通道
    SRAI_CHAN_ALARMIN, //告警输入通道

```

```

    SRAI_CHAN_ALARMOUT, //告警输出通道
    SRAI_CHAN_DECODE,    //解码通道
};

// 参数类型
enum
{
    SRAI_CMD_RESOURCECFG = 0x00000000, //监控资源参数
    SRAI_CMD_NETCFG, //网络参数
    SRAI_CMD_PICCFG, //图象参数
    SRAI_CMD_COMPRESSCFG, //压缩参数
    SRAI_CMD_RECORDCFG, //通道录像参数
    SRAI_CMD_RS232CFG, //232 串口参数
    SRAI_CMD_ALARMINCFG, //告警输入参数
    SRAI_CMD_TIMECFG, //监控资源时间
    SRAI_CMD_ZONEANDDST, //时区和夏时制参数
    SRAI_CMD_SHOWSTRING, //叠加字符参数
};

// 告警类型
enum
{
    SRAI_ALARM_CPU_HIGH = 0x00000000, //CPU 过高
    SRAI_ALARM_TEMPERATURE_ABNORMAL, //温度异常
    SRAI_ALARM_CLOCK_ABNORMAL, //时钟异常
    SRAI_ALARM_FAN_ABNORMAL, //风扇状态异常
    SRAI_ALARM_MEMORY_ABNORMAL, //内存状态异常
    SRAI_ALARM_DISK, //硬盘告警
    SRAI_ALARM_DISK_FULL, //硬盘满
    SRAI_ALARM_DISK_WRError, //硬盘读写出错
    SRAI_ALARM_VIDEO_LOST, //视频丢失
    SRAI_ALARM_VIDEO_ABNORMAL, //图形异常
    SRAI_ALARM_VIDEO_MISMATCH, //制式不匹配
    SRAI_ALARM_VIDEO_MOTION, //移动侦测
    SRAI_ALARM_VIDEO_COVERED, //视频遮挡
    SRAI_ALARM_INPUT, //告警输入
    SRAI_ALARM_ACCESS_OPEN, //门禁开
    SRAI_ALARM_ACCESS_CLOSE, //门禁关
    SRAI_ALARM_INFRARED, //红外告警
    SRAI_ALARM_OUTPUT, //告警输入参数
    SRAI_ALARM_NET, //网络异常
    SRAI_ALARM_SRAI_BROKEN, //网络断开
    SRAI_ALARM_IP_CONFLICT, //IP 地址冲突
};

```

```

    SRAI_ALARM_GPRS, //GPRS 通信异常
    SRAI_ALARM_3G, //3G 通信异常
    SRAI_ALARM_SIM_CARD, //SIM 卡异常
};

// 视频编码类型
enum
{
    VIDEO_NULL = 0x0000, //无视频
    VIDEO_H264, //H.264, 符合 ISO/IEC 14496.10(MPEG-4 AVC part 10;H.264)标准的规定
};

// 音频编码类型
enum
{
    AUDIO_NULL = 0x0000, //无音频
    AUDIO_AAC, //AAC 编码
};

// 帧类型
enum
{
    SRAI_FRAME_AUDIO_PCM = 0x0000, //音频帧,PCM
    SRAI_FRAME_VIDEO_I420, //I420
    SRAI_FRAME_VIDEO_YV12, //YV12
    SRAI_FRAME_VIDEO_NV12, //NV12
};

// 数据类型
enum
{
    SRAI_DATA_VIDEO = 0x0000, //视频数据
    SRAI_DATA_AUDIO, //音频数据
};

// 回放录像控制命令
typedef enum
{
    SRAI_PLAYBACK_START = 0x0000, //开始播放
    SRAI_PLAYBACK_PAUSE, //暂停播放
    SRAI_PLAYBACK_RESTART, //恢复播放
    SRAI_PLAYBACK_FAST, //快放
    SRAI_PLAYBACK_SLOW, //慢放
};

```



```

SRAI_PLAYBACK_NORMAL, //正常速度
SRAI_PLAYBACK_FRAME, //单帧放
SRAI_PLAYBACK_STARTAUDIO, //打开声音
SRAI_PLAYBACK_STOPAUDIO, //关闭声音
SRAI_PLAYBACK_AUDIOVOLUME, //调节音量 w
SRAI_PLAYBACK_SETPOS, //改变回放进度
SRAI_PLAYBACK_GETPOS, //获取回放进度
SRAI_PLAYBACK_GETTIME, //获取当前已经播放的时间
SRAI_PLAYBACK_GETFRAME, //获取当前已经播放的帧数
SRAI_PLAYBACK_GETTOTALFRAMES, //获取当前播放文件总的帧数
SRAI_PLAYBACK_GETTOTALTIME, //获取当前播放文件总的时间
SRAI_PLAYBACK_SETSPEED, //设置回放速度
SRAI_PLAYBACK_SETTIME, //按绝对时间定位
SRAI_PLAYBACK_FORWARD, //倒放切换为正放
SRAI_PLAYBACK_REVERSE, //正放切换为倒放
}SRAI_PLAYBACK_CMD;

```

// 云台控制类型

```
enum
```

```

{
    SRAI_PTZ_LIGHT_PWRON = 0x0002, //接通灯光电源
    SRAI_PTZ_WIPER_PWRON = 0x0003, //接通雨刷开关
    SRAI_PTZ_FAN_PWRON = 0x0004, //接通风扇开关
    SRAI_PTZ_HEATER_PWRON = 0x0005, //接通加热器开关
    SRAI_PTZ_AUX_PWRON1 = 0x0006, //接通辅助监控资源开关
    SRAI_PTZ_AUX_PWRON2 = 0x0007, //接通辅助监控资源开关
    SRAI_PTZ_SET_PRESET = 0x0008, //设置预置点
    SRAI_PTZ_CLE_PRESET = 0x0009, //清除预置点
    SRAI_PTZ_ZOOM_IN = 0x000A, //焦距变大
    SRAI_PTZ_ZOOM_OUT = 0x000B, //焦距变小
    SRAI_PTZ_FOCUS_NEAR = 0x000C, //焦点前调
    SRAI_PTZ_FOCUS_FAR = 0x000D, //焦点后调
    SRAI_PTZ_IRIS_OPEN = 0x000E, //光圈扩大
    SRAI_PTZ_IRIS_CLOSE = 0x000F, //光圈缩小
    SRAI_PTZ_UP = 0x0010, //上
    SRAI_PTZ_DOWN = 0x0011, //下
    SRAI_PTZ_LEFT = 0x0012, //左
    SRAI_PTZ_RIGHT = 0x0013, //右
    SRAI_PTZ_UP_LEFT = 0x0014, //左上
    SRAI_PTZ_UP_RIGHT = 0x0015, //右上
    SRAI_PTZ_DOWN_LEFT = 0x0016, //左下
    SRAI_PTZ_DOWN_RIGHT = 0x0017, //右下
    SRAI_PTZ_AUTO = 0x0018, //自动扫描

```

```

    SRAI_PTZ_FILL_PRE_SEQ = 0x0019, //将预置点加入巡航序列
    SRAI_PTZ_SET_SEQ_DWELL = 0x001A, //设置巡航点停顿时间
    SRAI_PTZ_SET_SEQ_SPEED = 0x001B, //设置巡航速度
    SRAI_PTZ_CLE_PRE_SEQ = 0x001C, //将预置点从巡航序列中删除
    SRAI_PTZ_STA_MEM_CRUISE = 0x001D, //开始记录轨迹
    SRAI_PTZ_STO_MEM_CRUISE = 0x001E, //停止记录轨迹
    SRAI_PTZ_RUN_CRUISE = 0x001F, //开始轨迹
    SRAI_PTZ_RUN_SEQ = 0x0020, //开始巡航
    SRAI_PTZ_STOP_SEQ = 0x0021, //停止巡航
    SRAI_PTZ_GOTO_PRESET = 0x0022, //快球转到预置点
    SRAI_PTZ_DEL_SEQ = 0x0023, //删除巡航路径
    SRAI_PTZ_PAN_CIRCLE = 0x0024, //云台自动圆周扫描
};

// 能力集类型
enum
{
    SRAI_CAPABILITY_BASICINFO = 0x0001, //设备基本信息
    SRAI_CAPABILITY_HSW, //设备软硬件
    SRAI_CAPABILITY_PTZ, //云台控制
};

// 操作回调函数
// 说明：异步调用的接口其操作结果会通过该接口注册的函数中返回
// 参数：dwOption[in]          操作类型
// lHandle[in]                 操作句柄
// pBuf[in]                    操作结果数据，由操作类型决定
// dwBufLen[in]                操作结果数据长度
// pUser[in]                   用户数据
// 返回值：无
typedef void (STDCALL *fOptionCallBack)(DWORD dwOption
                                         , LONG lHandle
                                         , char * pBuf
                                         , DWORD dwBufLen
                                         , void* pUser);

// 消息回调函数
// 说明：监控资源相关的业务消息会通过该接口注册的函数中返回
// 参数：dwCommand[in]         消息类型
// pBuf[in]                    消息数据，由消息类型决定
// dwBufLen[in]                消息数据长度
// pUser[in]                   用户数据
// 返回值：无

```

```

typedef void (STDCALL *fMessageCallBack)(DWORD dwCommand
                                         , char * pBuf
                                         , DWORD dwBufLen
                                         , void* pUser);

// 告警回调函数
// 说明：监控资源的告警数据会通过该接口注册的函数中返回
// 参数：dwAlarmType[in]      告警类型
// pAlarmInfo[in]            告警数据，由告警类型决定
// dwAlarmInfoLen[in]        告警数据长度
// pUser[in]                 用户数据
// 返回值：无
typedef void (STDCALL *fAlarmCallBack)(DWORD dwAlarmType
                                         , char * pAlarmInfo
                                         , DWORD dwAlarmInfoLen
                                         , void* pUser);

// 透明通道数据回调函数
// 说明：透明通道的数据会通过该接口注册的函数中返回
// 参数：IHandle [in]         透明通道句柄，SRAI_SerialStart 的返回值
// pBuf[in]                   透明通道数据
// dwBufLen[in]               数据长度
// pUser[in]                   用户数据
typedef void (STDCALL *fSerialDataCallBack)(LONG IHandle
                                             , CHAR * pBuf
                                             , DWORD dwBufLen
                                             , void* pUser);

#pragma pack(4)
typedef struct
{
    CHAR        sFactory[NAME_LEN];
    DWORD       dwVersion;
    CHAR        sSeries[SERIES_LEN];
}SRAI_INFO, *LPSRAI_INFO;

typedef struct
{
    DWORD       dwLogin;
    DWORD       dwRealplay;
    DWORD       dwPlayback;
    DWORD       dwUpgrade;
    DWORD       dwTalk;
    DWORD       dwSerial;
}

```

```
}SRAI_ABILITY, *LPSRAI_ABILITY;
```

```
typedef struct
```

```
{  
    DWORD    dwTotalLoginNum;  
    DWORD    dwTotalRealPlayNum;  
    DWORD    dwTotalPlayBackNum;  
    DWORD    dwTotalSerialNum;  
    DWORD    dwTotalUpgradeNum;  
    DWORD    dwTotalTalkNum;  
}SRAI_SRAISTATE, *LPSRAI_SRAISTATE;
```

```
typedef struct
```

```
{  
    DWORD    dwError;  
    CHAR     sErrorMsg[256];  
}SRAI_ERROR_INFO, *LPSRAI_ERROR_INFO;
```

```
typedef struct
```

```
{  
    DWORD    dwYear;  
    DWORD    dwMonth;  
    DWORD    dwDay;  
    DWORD    dwHour;  
    DWORD    dwMinute;  
    DWORD    dwSecond;  
    DWORD    dwMillisecond;  
}SRAI_TIME, *LPSRAI_TIME;
```

```
typedef struct
```

```
{  
    BYTE     byType;  
    BYTE     byRes[3];  
    CHAR     sIPv4[16];  
    BYTE     byIPv6[128];  
}SRAI_IPADDR, *LPSRAIIPADDR;
```

```
typedef struct
```

```
{  
    BYTE     byStartHour;  
    BYTE     byStartMin;  
    BYTE     byStopHour;  
    BYTE     byStopMin;
```

```
}SRAI_SCHEDTIME, *LPSRAI_SCHEDTIME;
```

```
typedef struct
```

```
{  
    DWORD    dwMonth;  
    DWORD    dwWeekNo;  
    DWORD    dwWeekDate;  
    DWORD    dwHour;  
    DWORD    dwMin;  
}SRAI_TIMEPOINT, *LPSRAI_TIMEPOINT;
```

```
typedef struct{
```

```
    WORD      wWidth;  
    WORD      wHeight;  
    DWORD     dwStamp;  
    WORD      wType;  
    WORD      nFrameRate;  
    DWORD     dwFrameNum;  
}SRAI_FRAME_INFO;
```

```
typedef struct
```

```
{  
    WORD      wChannel;  
    CHAR      sName[NAME_LEN];  
}SRAI_PICTURE, *LPSRAI_PICTURE;
```

```
typedef struct
```

```
{  
    DWORD     dwCount;  
    LPSRAI_PICTURE  Pictures;  
}SRAI_PICTURE_GROUP, *LPSRAI_PICTURE_GROUP;
```

```
typedef struct
```

```
{  
    DWORD     dwPicWidth;  
    DWORD     dwPicLength;  
    DWORD     dwPicQuality;  
}SRAI_JPEGPARA, *LPSRAI_JPEGPARA;
```

```
typedef struct
```

```
{  
    WORD      wChannel;  
    BYTE      byType;
```

```

        BYTE            byRes;
        CHAR            sName[NAME_LEN];
        DWORD           dwTotalSeconds;
        SRAI_TIME       StartTime;
        SRAI_TIME       EndTime;
    }SRAI_RECORD, *LPSRAI_RECORD;

typedef struct
{
    DWORD               dwCount;
    LPSRAI_RECORD       Records;
}SRAI_RECORD_GROUP, *LPSRAI_RECORD_GROUP;

typedef struct
{
    SRAI_TIME   SampleTime;
    DWORD       dwLongitude;
    DWORD       dwLatitude;
    BYTE        byType;
    BYTE        byRes[3];
    DWORD       dwDirection;
    DWORD       dwSpeed;
    DWORD       dwHeight;
}SRAI_GPS_INFO, *LPSRAI_GPS_INFO;
// 位置回调函数
// 说明： 监控资源的位置信息的数据会通过该接口注册的函数中返回
// 参数： GPS[in]                位置信息
// pUser[in]                    用户数据
// 返回值： 无
typedef void (STDCALL *fGPSCallBack)(LPSRAI_GPS_INFO GPS
                                     , void* pUser);

typedef struct
{
    DWORD       dwDataType;
    SRAI_FRAME_INFO   FrameInfo;
    DWORD       dwRecvSize;
    DWORD       dwBufSize;
    BYTE        *pBuffer;
}SRAI_REALDATA, *LPSRAI_REALDATA;
// 数据回调函数
// 说明： 实时预览和下载的数据会通过该接口注册的函数中返回
// 参数： IHandle[in]            数据句柄

```

```

// Data[in]                                回调数据，若原始码流的视频编码类型不在技术要求定义内，必须
回调为解码后数据
// pUser[in]                                用户数据
// 返回值：无
typedef void (STDCALL *fDataCallBack)(LONG lHandle
                                     , LPSRAI_REALDATA Data
                                     , void* pUser);

typedef struct
{
    CHAR        sSerialNumber[SERIALNO_LEN];
    DWORD       dwSoftwareVersion;
    DWORD       dwSoftwareBuildDate;
    DWORD       dwDSPSoftwareVersion;
    DWORD       dwDSPSoftwareBuildDate;
    DWORD       dwPanelVersion;
    DWORD       dwHardwareVersion;
    BYTE        byAlarmInNum;
    BYTE        byAlarmOutNum;
    BYTE        byRS232Num;
    BYTE        byRS485Num;
    BYTE        byNetworkPortNum;
    BYTE        byDiskCtrlNum;
    BYTE        byDiskNum;
    BYTE        byVideoChanNum;
    BYTE        byDecordChans;
    BYTE        byVGANum;
    BYTE        byUSBNum;
    BYTE        byAuxoutNum;
    BYTE        byAudioNum;
    BYTE        byIPChanNum;
    BYTE        byRes[2];
} SRAI_RESOURCECFG, *LPSRAI_RESOURCECFG;

typedef struct
{
    CHAR        sChanName[NAME_LEN];
    WORD        wChannel;
    BYTE        byType;
    BYTE        byStreamCount;
} SRAI_CHAN_INFO, *LPSRAI_CHAN_INFO;

typedef struct {

```

```

LONG                lSessionID;
CHAR                sName[NAME_LEN];
CHAR                sResourceID[NAME_LEN];
SRAI_IPADDR        RESOURCEIP;
SRAI_RESOURCECFG    RESOURCECf;
WORD               wFactory;
BYTE               byOnlineStatus;
BYTE               byBusType;
CHAR               sSeriesName[NAME_LEN];
CHAR               sArea[NAME_LEN];
BYTE               byGPSEnable;
BYTE               byRes1[3];
SRAI_GPS_INFO      GPS;
DWORD              dwChanNum;
LPSRAI_CHAN_INFO * Channels;
}SRAI_RESOURCE_INFO, *LPSRAI_RESOURCE_INFO;

```

```

typedef struct
{
    DWORD          dwHandleType;
    BYTE           byRelAlarmOut[MAX_ALARMOUT];
}SRAI_HANDLEALARM, *LPSRAI_HANDLEALARM;

```

```

typedef struct
{
    SRAI_TIME      AlarmTime;
    DWORD          dwAlarmType;
    BYTE           byReset;
    BYTE           byRes;
    WORD           wChannel;
    LPSRAI_PICTURE_GROUP    PictureGroup;
    LPSRAI_RECORD_GROUP     RecordGroup;
}SRAI_ALARM_INFO, *LPSRAI_ALARM_INFO;

```

```

typedef struct
{
    DWORD          dwPPPOE;
    CHAR           sPPPoEUser[NAME_LEN];
    CHAR           sPPPoEPassword[PASSWD_LEN];
    SRAI_IPADDR    struPPPoEIP;
}SRAI_PPPOECFG, *LPSRAI_PPPOECFG;

```

```

typedef struct

```



```

{
    SRAI_IPADDR      struIP;
    SRAI_IPADDR      struIPMask;
    WORD             wMTU;
    BYTE             byUseDhcp;
    BYTE             byRes;
    BYTE             byMACAddr[MACADDR_LEN];
    SRAI_IPADDR      struGatewayIpAddr;
}SRAI_ETHERNET, *LPSRAI_ETHERNET;

```

typedef struct

```

{
    BYTE             byDefaultRoute;
    BYTE             byNetworkCardNum;
    BYTE             byRes[2];
    SRAI_ETHERNET    struEtherNet[MAX_NETWORK_CARD];
    BYTE             byIpResolver[MAX_DOMAIN_NAME];
    WORD             wIpResolverPort;
    WORD             wLocalPort;
    WORD             wHttpPortNo;
    BYTE             byRes1[2];
    SRAI_IPADDR      struMulticastIpAddr;
    SRAI_PPPOECFG    struPPPoE;
}SRAI_NETCFG, *LPSRAI_NETCFG;

```

typedef struct

```

{
    WORD            wHideAreaTopLeftX;
    WORD            wHideAreaTopLeftY;
    WORD            wHideAreaWidth;
    WORD            wHideAreaHeight;
}SRAI_SHELTER, *LPSRAI_SHELTER;

```

typedef struct

```

{
    BYTE             byEnableHandleVILost;
    BYTE             byRes[3];
    SRAI_HANDLEALARM strVILostHandleType;
    SRAI_SCHEDULETIME struAlarmTime[MAX_DAYS][8];
}SRAI_VIDEO_LOST, *LPSRAI_VIDEO_LOST;

```

typedef struct

```

{

```

```

        DWORD                dwEnableHideAlarm;
        WORD                 wHideAlarmAreaTopLeftX;
        WORD                 wHideAlarmAreaTopLeftY;
        WORD                 wHideAlarmAreaWidth;
        WORD                 wHideAlarmAreaHeight;
        SRAI_HANDLEALARM     strHideAlarmHandleType;
        SRAI_SCHEDTIME        struAlarmTime[MAX_DAYS][8];
}SRAI_VIDEO_HIDE, *LPSRAI_VIDEO_HIDE;

```

typedef struct

```

{
    BYTE                byMotionScope[64][96];
    BYTE                byMotionSensitive;
    BYTE                byEnableHandleMotion;
    BYTE                byEnableDisplay;
    BYTE                byRes;
    SRAI_HANDLEALARM     struMotionHandleType;
    SRAI_SCHEDTIME        struAlarmTime[MAX_DAYS][8];
    BYTE                byRelRecordChan[MAX_CHANNUM];
}SRAI_VIDEO_MOTION, *LPSRAI_VIDEO_MOTION;

```

typedef struct

```

{
    WORD                wChannel;
    BYTE                byRes[2];
    DWORD               dwVideoFormat;
    DWORD               dwShowChanName;
    WORD                wShowNameTopLeftX;
    WORD                wShowNameTopLeftY;
    SRAI_VIDEO_LOST     struVILost;
    SRAI_VIDEO_MOTION   struMotion;
    SRAI_VIDEO_HIDE     struHideAlarm;
    DWORD               dwEnableHide;
    SRAI_SHELTER         struShelter[MAX_SHELTERNUM];
    DWORD               dwShowOsd;
    WORD                wOSDTopLeftX;
    WORD                wOSDTopLeftY;
    BYTE                byOSDType;
    BYTE                byDispWeek;
    BYTE                byOSDAtrib;
    BYTE                byHourOSDType;
    BYTE                byFontSize;
    BYTE                byRes1[3];
}

```

```
}SRAI_PICCFG, *LPSRAI_PICCFG;
```

```
typedef struct
```

```
{  
    WORD        wChannel;  
    BYTE        byStreamNO;  
    BYTE        byStreamType;  
    WORD        wWidth;  
    WORD        wHeight;  
    BYTE        byBitrateType;  
    BYTE        byPicQuality;  
    BYTE        byRes[2];  
    DWORD       dwVideoBitrate;  
    DWORD       dwVideoFrameRate;  
    WORD        wIntervalFrameI;  
    BYTE        byIntervalBPFrame;  
    BYTE        byRes1;  
    WORD        wVideoEncType;  
    WORD        wAudioEncType;  
    BYTE        byVideoEncComplexity;  
    BYTE        byRes2[3];  
}SRAI_COMPRESSIONCFG, *LPSRAI_COMPRESSIONCFG;
```

```
typedef struct
```

```
{  
    WORD        wAllDayRecord;  
    BYTE        byRecordType;  
    BYTE        byRes;  
}SRAI_RECORDDAY, *LPSRAI_RECORDDAY;
```

```
typedef struct
```

```
{  
    SRAI_SCHEDTIME    struRecordTime;  
    BYTE              byRecordType;  
    BYTE              byRes[3];  
}SRAI_RECORDSCHED, *LPSRAI_RECORDSCHED;
```

```
typedef struct
```

```
{  
    WORD        wChannel;  
    BYTE        byRes[2];  
    DWORD       dwRecord;  
    SRAI_RECORDDAY    struRecAllDay[MAX_DAYS];  
}
```

```

        SRAI_RECORDSCHED    struRecordSched[MAX_DAYS][8];
        DWORD               dwRecordTime;
        DWORD               dwPreRecordTime;
        DWORD               dwRecorderDuration;
        BYTE                byRedundancyRec;
        BYTE                byAudioRec;
        BYTE                byRes1[2];
    } SRAI_RECORDCFG, *LPSRAI_RECORDCFG;

```

typedef struct

```

{
    WORD                wChannel;
    BYTE                byRes[2];
    DWORD               dwBaudRate;
    BYTE                byDataBit;
    BYTE                byStopBit;
    BYTE                byParity;
    BYTE                byFlowcontrol;
    DWORD               dwWorkMode;
} SRAI_RS232CFG, *LPSRAI_RS232CFG;

```

typedef struct

```

{
    CHAR                sAlarmInName[NAME_LEN];
    BYTE                byAlarmType;
    BYTE                byAlarmInHandle;
    WORD                wChannel;
    SRAI_HANDLEALARM    struAlarmHandleType;
    SRAI_SCHEDULETIME   struAlarmTime[MAX_DAYS][8];
    BYTE                byRelRecordChan[MAX_CHANNUM];
    BYTE                byEnablePreset[MAX_CHANNUM];
    BYTE                byPresetNo[MAX_CHANNUM];
    BYTE                byEnableCruise[MAX_CHANNUM];
    BYTE                byCruiseNo[MAX_CHANNUM];
    BYTE                byEnablePtzTrack[MAX_CHANNUM];
    BYTE                byPTZTrack[MAX_CHANNUM];
} SRAI_ALARMINCFG, *LPSRAI_ALARMINCFG;

```

typedef struct

```

{
    DWORD               dwEnableDST;
    BYTE                byDSTBias;
    BYTE                byRes[3];
}

```

```

        SRAI_TIMEPOINT    struBeginPoint;
        SRAI_TIMEPOINT    struEndPoint;
    }SRAI_ZONEANDDST, *LPSRAI_ZONEANDDST;

```

```

typedef struct
{
    WORD            wShowString;
    WORD            wStringSize;
    WORD            wShowStringTopLeftX;
    WORD            wShowStringTopLeftY;
    CHAR            sString[44];
}SRAI_SHOWSTRINGINFO, *LPSRAI_SHOWSTRINGINFO;

```

```

typedef struct
{
    WORD            wChannel;
    BYTE            byRes[2];
    SRAI_SHOWSTRINGINFO    struStringInfo[MAX_STRINGNUM];
}SRAI_SHOWSTRING, *LPSRAI_SHOWSTRING;

```

```

typedef struct {
    SRAI_IPADDR    IP;
    WORD            wPort;
    BYTE            byRes[2];
    CHAR            sName[NAME_LEN];
    CHAR            sPassword[PASSWD_LEN];
    WORD            wReconnectTime;
    BYTE            byRes1[2];
}SRAI_LOGIN_INFO, *LPSRAI_LOGIN_INFO;

```

```

typedef struct {
    SRAI_IPADDR    IP;
    WORD            wPort;
    BYTE            byRes[2];
}SRAI_START_SERVER, *LPSRAI_START_SERVER;

```

```

typedef struct
{
    DWORD            dwType;
    DWORD            dwBufLen;
    BYTE*            pBuf;
}SRAI_PARAM, *LPSRAI_PARAM;

```

```
typedef struct
{
    WORD            wChannel;
    BYTE            byStreamNO;
    BYTE            byLinkMode;
    fDataCallBack   DataCBK;
    void*           pUser;
}SRAI_REALPLAY, *LPSRAI_REALPLAY;
```

```
typedef struct
{
    WORD            wChannel;
    BYTE            byType;
    BYTE            byRes;
    CHAR            sName[NAME_LEN];
    DWORD           dwStartPos;
    SRAI_TIME       StartTime;
    SRAI_TIME       EndTime;
    fDataCallBack   DataCBK;
    void*           pUser;
}SRAI_PLAYBACK, *LPSRAI_PLAYBACK;
```

```
typedef struct
{
    DWORD           dwCmd;
    BYTE*           pInBuffer;
    DWORD           dwInLen;
    BYTE*           pOutBuffer;
    DWORD           dwOutLen;
}SRAI_PLAYBACK_CONTROL, *LPSRAI_PLAYBACK_CONTROL;
```

```
typedef struct
{
    WORD            wChannel;
    BYTE            byType;
    BYTE            byRes;
    CHAR            sName[NAME_LEN];
    DWORD           dwBufLen;
    BYTE*           pBuf;
}SRAI_SNAPSHOT_PARAM, *LPSRAI_SNAPSHOT_PARAM;
```

```
typedef struct
{
```

```

        WORD            wChannel;
        WORD            wCommand;
        BYTE            byStop;
        BYTE            byRes[3];
        DWORD           dwSpeed;
        DWORD           dwPresetNO;
        BYTE            byCruiseRoute;
        BYTE            byCruisePoint;
        BYTE            byRes1[2];
        DWORD           dwTime;
}SRAI_PTZ_CTRL, *LPSRAI_PTZ_CTRL;

```

```

typedef struct
{
        WORD            wChannel;
        BYTE            byRes[2];
}SRAI_START_RECORD, *LPSRAI_START_RECORD;

```

```

typedef struct
{
        WORD            wChannel;
        BYTE            byRes[2];
        SRAI_TIME        StartTime;
        SRAI_TIME        EndTime;
}SRAI_QUERY, *LPSRAI_QUERY;

```

```

typedef struct
{
        WORD            wChannel;
        BYTE            byRes[2];
        fDataCallBack    DataCBK;
        void*            pUser;
} SRAI_TALK, * LPSRAI_TALK;

```

```

typedef struct{
        SRAI_TIME        LogTime;
        DWORD            dwType;
        CHAR              sPanelUser[NAME_LEN];
        CHAR              sNetUser[NAME_LEN];
        CHAR              sRemoteHostAddr[16];
        DWORD            dwChannel;
        DWORD            dwDiskNumber;
        DWORD            dwAlarmInPort;

```

```

        DWORD          dwAlarmOutPort;
}SRAI_LOG,*LPSRAI_LOG;

#pragma pack()

// 通用

// 初始化环境
// 说明: 初始化 SRAI 环境, 在其他接口之前调用
// 参数: CBK[in]          操作回调函数
// pUser[in]             用户数据, 在操作回调函数中带回
// 返回值: >=0           成功
//          -1           失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_Initialize(fOptionCallBack CBK
                                     , void* pUser);

// 清除环境
// 说明: 清除 SRAI 环境, 并终止所有操作
// 参数: 无
// 返回值: >=0           成功
//          -1           失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_Finalize();

// 设置消息回调函数
// 说明: 设置消息回调函数, 监控资源相关的业务消息会通过该接口设置的函数返回
// 参数: CBK[in]          消息回调函数
// pUser[in]             用户数据, 在消息回调函数中带回
// 返回值: >=0           成功
//          -1           失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_SetMsgCallback(fMessageCallBack CBK
                                     , void* pUser);

// 登录监控资源
// 说明: 登录监控资源, 返回用户句柄
// 参数: Info [in]        连接信息
// 返回值: >=0           成功, 返回用户句柄
//          -1           失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_Login(LPSRAI_LOGIN_INFO Info);

// 登出监控资源
// 说明: 根据用户句柄登出监控资源
// 参数: IUserID [in]      用户句柄, SRAI_Login 返回值
// 返回值: >=0           成功

```



```

//          -1                      失败，获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_Logout(LONG IUserID);

// 开启注册服务
// 说明：开启注册服务供车载监控资源等移动监控资源进行注册，并返回用户句柄
// 参数：Info [in]                  连接信息
// 返回值：>=0                      成功，返回用户句柄
//          -1                      失败，获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_StartServer (LPSRAI_START_SERVER Info);

// 关闭注册服务
// 说明：根据用户句柄关闭注册服务
// 参数：IUserID [in]               用户句柄，SRAI_StartServer 返回值
// 返回值：>=0                      成功
//          -1                      失败，获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_StopServer (LONG IUserID);

// 获取 SRAI 能力集
// 说明：返回 SRAI 能力集
// 参数：pSRAIAbl [out]             功能信息结构
// 返回值：>=0                      成功
//          -1                      失败，获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_GetSRAICapability(LPSRAI_ABILITY pSRAIAbl);

// 获取监控资源能力集
// 说明：返回 SRAI 监控资源能力集
// 参数：IUserID [in]               用户句柄，SRAI_Login 或 SRAI_StartServer 返回值
// bType[in]                       能力集类型
// pCapabilityBuf[in]               能力集缓冲区指针
// dwCapabilityLength[in,out]       能力集缓冲区的长度
// 返回值：>=0                      成功
//          -1                      失败，获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_GetResourceCapability(LONG IUserID
                                                , BYTE type
                                                , char *pCapabilityBuf
                                                , DWORD dwCapabilityLength);

// 获取 SRAI 状态
// 说明：返回 SRAI 状态
// 参数：pSRAIState [out]           状态信息结构
// 返回值：>=0                      成功
//          -1                      失败，获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_GetSRAIState(LPSRAI_SRAI_STATE pSRAIState);

```

```

// 获取 SRAI 信息
// 说明: 返回 SRAI 信息
// 参数: pSRAIInfo [out]          SRAI 信息结构
// 返回值: >=0                    成功
//          -1                    失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_GetInfo(LPSRAI_INFO pSRAIInfo);

// 获取最后操作的错误信息
// 说明: 返回最后一次操作的错误信息
// 参数: pError [out]             SRAI 错误信息
// 返回值: 无
SRAI_API void STDCALL SRAI_GetLastError(LPSRAI_ERROR_INFO pError);

// 获取日志
// 说明: 返回日志句柄
// 参数: IUserID[in]              用户句柄, SRAI_Login 或 SRAI_StartServer 返回值
//          ISelectMode[in]        查询方式
//          dwType[in]             日志类型
//          StartTime[in]          开始时间
//          StopTime[in]           停止时间
// 返回值: >=0                    成功, 返回日志查询句柄
//          -1                    失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_FindLog(LONG ISession
                                   , LONG ISelectMode
                                   , DWORD dwType
                                   , LPSRAI_TIME StartTime
                                   , LPSRAI_TIME StopTime);

// 关闭日志查询句柄
// 说明: 关闭日志查询句柄
// 参数: IHandle[in]              日志查询句柄, SRAI_FindLog 返回值
// 返回值: >=0                    成功
//          -1                    失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_CloseFindLog(LONG IHandle);

// 逐条获取日志信息
// 说明: 逐条获取日志信息
// 参数: IHandle[in]              日志查询句柄, SRAI_FindLog 返回值
//          Log[out]              日志信息
// 返回值: >=0                    成功
//          -1                    失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_FindNextLog(LONG IHandle
                                       , LPSRAI_LOG Log);

```

```

// 媒体

// 开启实时预览
// 说明：开启实时预览，返回实时预览句柄
// 参数：ISessionID [in]          监控资源会话 ID，SRAI_RESOURCE_INFO 中的 ISessionID
// RealPlay [in]                  播放参数
// 返回值：>=0                    成功，返回播放句柄
//          -1                    失败，获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_RealPlay(LONG ISession
                                     , LPSRAI_REALPLAY RealPlay);

// 停止实时预览
// 说明：根据实时预览句柄停止实时预览
// 参数：IHandle [in]             播放句柄，SRAI_RealPlay 返回值
// 返回值：>=0                    成功
//          -1                    失败，获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_StopRealPlay(LONG IHandle);

// 查询图片
// 说明：查询图片，结果由消息回调函数返回
// 参数：ISessionID [in]          监控资源会话 ID，SRAI_RESOURCE_INFO 中的 ISessionID
// Query [in]                     图片查询参数
// 返回值：>=0                    成功，返回查询句柄
//          -1                    失败，获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_QueryPicture(LONG ISession
                                         , LPSRAI_QUERY Query);

// 取消查询图片
// 说明：查询图片，结果由消息回调函数返回
// 参数：IHandle [in]             查询句柄，SRAI_QueryPicture 返回值
// 返回值：>=0                    成功
//          -1                    失败，获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_CancelQueryPicture(LONG IHandle);

// 下载图片
// 说明：下载图片，返回图片下载句柄
// 参数：ISessionID [in]          监控资源会话 ID，SRAI_RESOURCE_INFO 中的 ISessionID
// Picture [in]                   图片信息
// pBuf [in,out]                  图片数据
// dwBufLen [in,out]              输入 pBuf 长度，输出图片数据长度
// 返回值：>=0                    成功，返回下载句柄
//          -1                    失败，获取错误码调用 SRAI_GetLastError

```

```

SRAI_API LONG STDCALL SRAI_DownloadPicture(LONG lSession
                                           , LPSRAI_PICTURE Picture
                                           , BYTE* pBuf
                                           , DWORD dwBufLen);

// 查询录像
// 说明: 查询监控资源录像, 结果通过消息回调函数返回
// 参数: lSessionID [in]      监控资源会话 ID, SRAI_RESOURCE_INFO 中的 lSessionID
// Query [in]                录像查询参数
// 返回值: >=0               成功, 返回查询句柄
//          -1                失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_QueryRecord(LONG lSession
                                       , LPSRAI_QUERY Query);

// 取消查询录像
// 说明: 取消监控资源录像, 结果通过消息回调函数返回
// 参数: lHandle [in]         查询句柄, SRAI_QueryRecord 返回值
// 返回值: >=0               成功
//          -1                失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_CancelQueryRecord(LONG lHandle);

// 回放录像
// 说明: 回放监控资源录像, 并返回回放句柄
// 参数: lSessionID [in]      监控资源会话 ID, SRAI_RESOURCE_INFO 中的 lSessionID
// Record [in]                录像回放信息
// 返回值: >=0               成功, 返回回放句柄
//          -1                失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_PlayBack(LONG lSession
                                    , LPSRAI_PLAYBACK Record);

// 停止回放录像
// 说明: 根据回放句柄停止回放录像
// 参数: lHandle [in]         回放句柄, SRAI_PlayBack 返回值
// 返回值: >=0               成功
//          -1                失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_StopPlayBack(LONG lHandle);

// 下载录像
// 说明: 下载监控资源录像, 并返回下载句柄
// 参数: lSessionID [in]      监控资源会话 ID, SRAI_RESOURCE_INFO 中的 lSessionID
// Record [in]                录像回放下载信息
// 返回值: >=0               成功, 返回回放句柄
//          -1                失败, 获取错误码调用 SRAI_GetLastError

```

```

SRAI_API LONG STDCALL SRAI_Download(LONG ISession
                                     , LPSRAI_PLAYBACK Record);

// 停止下载录像
// 说明: 根据下载句柄停止下载录像
// 参数: IHandle [in]           下载句柄, SRAI_Download 返回值
// 返回值: >=0                 成功
//          -1                  失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_StopDownload(LONG IHandle);

// 回放录像控制
// 说明: 根据回放句柄控制回放录像
// 参数: IHandle [in]           回放句柄, SRAI_PlayBack 返回值
// Control [in]                 回放录像控制参数
// 返回值: >=0                 成功
//          -1                  失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_PlayBackControl(LONG IHandle
                                             , LPSRAI_PLAYBACK_CONTROL Control);

// 打开语音对接
// 说明: 开启和监控资源的语音对接, 并返回语音对接句柄
// 参数: ISessionID [in]        监控资源会话 ID, SRAI_RESOURCE_INFO 中的 ISessionID
// talk [in]                    对讲参数
// 返回值: >=0                 成功, 返回对讲句柄
//          -1                  失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_StartTalk(LONG ISession
                                     , LPSRAI_TALK talk);

// 停止语音对接
// 说明: 根据语音对接句柄停止语音对接
// 参数: IHandle [in]           对讲句柄, SRAI_StartTalk 返回值
// 返回值: >=0                 成功
//          -1                  失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_StopTalk(LONG IHandle);

// 发送语音数据
// 说明: 根据语音对接句柄发送语音数据
// 参数: IHandle [in]           对讲句柄, SRAI_StartTalk 返回值
// pBuf[in]                     语音数据
// dwBufLen                     语音数据长度
// 返回值: >=0                 成功
//          -1                  失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_SendTalk(LONG IHandle

```

```

        , char* pBuf
        , DWORD dwBufLen);

// 抓拍
// 说明: 调用监控资源抓图, 并返回图像信息
// 参数: lSessionID [in]      监控资源会话 ID, SRAI_RESOURCE_INFO 中的 lSessionID
// Snapshot [in]            抓拍参数
// 返回值: >=0              成功
//          -1              失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_Snapshot(LONG lSession
                                     , LPSRAI_SNAPSHOT_PARAM Snapshot);

// 手动录像
// 说明: 手动开启监控资源录像
// 参数: lSessionID [in]      监控资源会话 ID, SRAI_RESOURCE_INFO 中的 lSessionID
// StartRec [in]             手动录像参数
// 返回值: >=0              成功
//          -1              失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_StartRecord(LONG lSession
                                       , LPSRAI_START_RECORD StartRecr);

// 停止录像
// 说明: 手动停止监控资源录像
// 参数: lSessionID [in]      监控资源会话 ID, SRAI_RESOURCE_INFO 中的 lSessionID
// StopRec [in]              手动录像参数
// 返回值: >=0              成功
//          -1              失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_StopRecord(LONG lSession
                                       , LPSRAI_START_RECORD StopRec);

// 云台控制
// 说明: 根据控制命令不同进行不同的云台控制, 进行控制时不需开启实时预览
// 参数: lSessionID [in]      监控资源会话 ID, SRAI_RESOURCE_INFO 中的 lSessionID
// PTZ [in]                  云台控制命令
// 返回值: >=0              成功
//          -1              失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_PTZControl(LONG lSession
                                       , LPSRAI_PTZ_CTRL PTZ);

// 监控资源维护

// 重启监控资源
// 说明: 重启监控资源

```

```

// 参数: lSessionID [in]          监控资源会话 ID, SRAI_RESOURCE_INFO 中的 lSessionID
// 返回值: >=0                    成功
//          -1                    失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_RebootResource(LONG    lSession);

// 远程升级
// 说明: 使用本地文件的方式进行监控资源的远程升级, 并返回远程升级句柄
// 参数: lSessionID [in]          监控资源会话 ID, SRAI_RESOURCE_INFO 中的 lSessionID
// sName [in]                    升级的文件路径, 包括文件名
// 返回值: >=0                    成功, 返回远程升级句柄
//          -1                    失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_Upgrade(LONG    lSession
                                   , CHAR*   sName);

// 获取远程升级的进度
// 说明: 根据远程升级返回的句柄获取远程升级的进度
// 参数: lHandle [in]            远程升级句柄, SRAI_Upgrade 返回值
// 返回值: >=0                    成功
//          -1                    失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_GetUpgradeProgress(LONG    lHandle);

// 获取远程升级的状态
// 说明: 根据远程升级返回的句柄获取远程升级的状态
// 参数: lHandle [in]            远程升级句柄, SRAI_Upgrade 返回值
// lState[out]                    远程升级状态, 定义如下:
//                                -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
//                                6- 升级写 flash 失败
// 返回值: >=0                    成功
//          -1                    失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_GetUpgradeState(LONG    lHandle
                                           , LONG*   lState);

// 终止远程升级
// 说明: 根据远程升级返回的句柄终止远程升级
// 参数: lHandle [in]            远程升级句柄, SRAI_Upgrade 返回值
// 返回值: >=0                    成功
//          -1                    失败, 获取错误码调用 SRAI_GetLastError

```

```

SRAI_API LONG STDCALL SRAI_StopUpgrade(LONG    IHandle);

// 参数配置

// 获取参数
// 说明: 获取监控资源参数, 根据参数类型进行获取
// 参数: ISessionID [in]          监控资源会话 ID, SRAI_RESOURCE_INFO 中的 ISessionID
// Param [in]                    参数信息
// 返回值: >=0                    成功
//          -1                    失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_GetParam(LONG ISession
                                     , LPSRAI_PARAM Param);

// 设置参数
// 说明: 设置监控资源参数, 根据参数类型进行设置
// 参数: ISessionID [in]          监控资源会话 ID, SRAI_RESOURCE_INFO 中的 ISessionID
// Param [in,out]                参数信息
// 返回值: >=0                    成功
//          -1                    失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_SetParam(LONG ISession
                                     , LPSRAI_PARAM Param);

// 透明通道
// 建立透明通道
// 说明: 建立和监控资源间的透明通道并返回通道句柄
// 参数: ISessionID [in]          监控资源会话 ID, SRAI_RESOURCE_INFO 中的 ISessionID
// ISerialPort [in]              串口号: 1—232 串口; 2—485 串口
// CBK [in]                      回调函数
// pUser [in]                    用户数据
// 返回值: >=0                    成功, 返回透明通道句柄
//          -1                    失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_SerialStart(LONG    ISession
                                       , LONG    ISerialPort
                                       , fSerialDataCallBack CBK
                                       , void* pUser);

// 断开透明通道
// 说明: 根据建立透明通道时返回的句柄断开透明通道
// 参数: IHandle [in]            透明通道句柄, SRAI_SerialStart 返回值
// 返回值: >=0                    成功
//          -1                    失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_SerialStop(LONG    IHandle);

```



```

// 通过透明通道向监控资源串口发送数据
// 说明: 根据建立透明通道时返回的句柄向监控资源的串口发送数据
// 参数: IHandle [in]          透明通道句柄, SRAI_SerialStart 返回值
// wChannel[in]              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
// pSendBuf[in]             发送数据的缓冲区指针
// dwBufSize[in]            缓冲区的大小, 最多 1016 字节
// 返回值: >=0              成功
//          -1              失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_SerialSend(LONG      IHandle
                                     , WORD      wChannel
                                     , BYTE      *pSendBuf
                                     , DWORD      dwBufSize);

// 直接向 232 串口发送数据
// 说明: 直接向监控资源的 232 串口发送数据
// 参数: ISessionID [in]      监控资源会话 ID, SRAI_RESOURCE_INFO 中的 ISessionID
// pSendBuf[in]             发送数据的缓冲区指针
// dwBufSize[in]            缓冲区的大小, 最多 1016 字节
// 返回值: >=0              成功
//          -1              失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_SendTo232Port (LONG   ISession
                                     , BYTE      *pSendBuf
                                     , DWORD      dwBufSize);

// 直接向串口发送数据
// 说明: 直接向监控资源的串口发送数据
// 参数: ISessionID [in]      监控资源会话 ID, SRAI_RESOURCE_INFO 中的 ISessionID
// dwSerialPort[in]          串口类型: 1-232, 2-485
// dwSerialIndex[in]         表示第几个 232 或者 485, 从 1 开始
// pSendBuf[in]             发送数据的缓冲区指针
// dwBufSize[in]            缓冲区的大小, 最多 1016 字节
// 返回值: >=0              成功
//          -1              失败, 获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_SendToSerialPort (LONG   ISession
                                     , DWORD      dwSerialPort
                                     , DWORD      dwSerialIndex
                                     , BYTE      *pSendBuf
                                     , DWORD      dwBufSize);

// 设置告警回调函数
// 说明: 调用该接口后, 所有监控资源产生的告警信息通过回调函数返回, 默认为全布防, 取消设置为撤防
// 参数: CBK[in]            告警回调函数, 为 0 表示取消设置

```

```

// pUser[in]                用户数据，在告警回调函数中带回
// 返回值: >=0              成功
//          -1              失败，获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_SetAlarmCallBack(fAlarmCallBack CBK
                                           , void* pUser);

// 设置位置信息回调函数
// 说明：调用该接口后，所有监控资源的位置信息通过回调函数返回
// 参数: CBK[in]            位置信息回调函数，为 0 表示取消设置
// pUser[in]                用户数据，在位置信息回调函数中带回
// 返回值: >=0              成功-1
//          -1              失败，获取错误码调用 SRAI_GetLastError
SRAI_API LONG STDCALL SRAI_SetGPSCallBack(fGPSCallBack CBK
                                           , void* pUser);

#endif

```

附录 E  
(规范性附录)  
监控资源能力集

## E.1 基本信息

```
<?xml version="1.0" encoding="utf-8" ?>
<BasicInformation>
  <SerialNumber>XX-YYYY-0120140507827495792379</ SerialNumber>
  <SoftwareVersion>0x80010201</ SoftwareVersion>
  <SoftwareBuildDate>2014-01-01</ SoftwareBuildDate>
  <DSPSoftwareVersion>0x60220513</ DSPSoftwareVersion>
  <DSPSoftwareBuildDate>2013-01-01</ DSPSoftwareBuildDate>
  <PanelVersion>0x50010211</ PanelVersion>
</BasicInformation>
```

## E.2 软硬件能力集

```
<?xml version="1.0" encoding="utf-8" ?>
<BasicCapability>
<HardwareCapability>
  <AlarmInPortNum>4</AlarmInPortNum>
  <AlarmOutPortNum>4</AlarmOutPortNum>
  <RS232Num>1</RS232Num>
  <RS485Num>1</RS485Num>
  <NetworkPortNum>1</NetworkPortNum>
  <VideoInNum>4</VideoInNum>
  <AudioInNum>4</AudioInNum>
  <VideoOutNum>1</VideoOutNum>
  <AudioOutNum>1</AudioOutNum>
  <AudioTalkNum>2</AudioTalkNum>
</HardwareCapability>
<SoftwareCapability>
  <ShowStringNumber>8</ShowStringNumber>
  <MotionDetectAlarmSupport>1</MotionDetectAlarmSupport>
  <VILostAlarmSupport>1</VILostAlarmSupport>
  <HideAlarmSupport>1</HideAlarmSupport>
  <ShelterSupport>1</ShelterSupport>
  <EptzSupport>1</EptzSupport>
  <PtzSupport>1</PtzSupport>
  <MultipleStreamSupport>1</MultipleStreamSupport>
  <SubStreamSupport>1</SubStreamSupport>
```

```

<BitrateLimitSupport>1</BitrateLimitSupport>
<MaxLoginNum>0</MaxLoginNum>
<MaxPreviewNum>0</MaxPreviewNum>
<MaxPlayBackNum>0</MaxPlayBackNum>
<MaxChanLinkNum>6</MaxChanLinkNum>
<RecordSupport>1</ RecordSupport>
<PlayBackSupport>1</ PlayBackSupport>
<ShutDownSupport>1</ShutDownSupport>
<DownloadSupport>1</ DownloadSupport>
<VideoH264>1</ VideoH264>
<AudioAAC>1</ AudioAAC>
</SoftwareCapability>
</BasicCapability>

```

### E.3 云台控制能力集

```

<?xml version="1.0" encoding="utf-8" ?>
<PTZAbility>
<PTZControl>
<controlType
    opt="light, wiper, fan, heater, aux1, aux2, zoomIn, zoomOut, focusNear, focusFar, irisOpen, irisClose, ttiltUp,
    tiltDown, panLeft, panRight, upLeft, upRight, downLeft, downRight, panAuto, panCircle" />
</PTZControl>
<Patrol>
    <patrolNum min="1" max="16" />
    <presetNum min="1" max="255" />
    <dwelTime min="1" max="255" />
    <speed min="1" max="255" />
</Patrol>
<Preset>
    <presetNum min="1" max="255" />
    <nameLen max="32" />
    <specialNo opt="1, 2, 3, 4, 5" />
</Preset>
</PTZAbility>

```

### E.4 设备配置能力集

```

<?xml version="1.0" encoding="utf-8" ?>
<ConfigSet>
    <SNMPCfg>
        <SNMPV1Enabled> <!-- ro, req, xs:boolean ,是否启用 SNMP V1--> </SNMPV1Enabled>
        <SNMPV2Enabled> <!-- ro, req, xs:boolean,是否启用 SNMP V2 --> </SNMPV2Enabled>
    
```

```

<SNMPV3Enabled> <!-- ro, req, xs:boolean, 是否启用 SNMP V3--> </SNMPV3Enabled>
<ServerPort> <!-- ro, req, xs:integer, SNMP 消息接收端口 --> </ServerPort>
<ReadCommunity> <!-- req, xs:string, 读共同体 --> </ReadCommunity>
<WriteCommunity> <!-- req, xs:string, 写共同体 --> </WriteCommunity>
<TrapHostIP> <!-- req, xs:string, 自陷主机 IP 地址 --> </TrapHostIP>
<TrapHostPort> <!-- ro, req, xs:integer, 自陷主机端口 --> </TrapHostPort>
<RWUser> //读写用户
    <UserName> <!-- req, xs:string, 用户名 --> </UserName>
    <SecLevel> <!-- req, xs:string, "noParity, noAuthorizationCheck, authorizationCheck", 安全级别,
noParity-无校验, noAuthorizationCheck-无授权校验, authorizationCheck-授权校验 --> </SecLevel>
    <AuthType> <!-- req, xs:string, "none, MD5, SHA", 认证类型, none-无--> </AuthType>
    <AuthPassword> <!-- req, xs:string, 认证密码 --> </AuthPassword>
    <EncryptionType> <!-- req, xs:string, "none, DES, AES", 加密类型--> </EncryptionType>
    <EncryptionPassword> <!-- req, xs:string, 加密密码--> </EncryptionPassword>
</RWUser>
<ROUser> //只读用户
    <UserName> <!-- req, xs:string, 用户名--> </UserName>
    <SecLevel> <!-- req, xs:string, "noParity, noAuthorizationCheck, authorizationCheck", 安全级别,
noParity-无校验, noAuthorizationCheck-无授权校验, authorizationCheck-授权校验--> </SecLevel>
    <AuthType> <!-- req, xs:string, "none, MD5, SHA", 认证类型, none-无--> </AuthType>
    <AuthPassword> <!-- req, xs:string, 认证密码 --> </AuthPassword>
    <EncryptionType> <!-- req, xs:string, "none, DES, AES", 加密类型--> </EncryptionType>
    <EncryptionPassword> <!-- req, xs:string, 加密密码--> </EncryptionPassword>
</ROUser>
</SNMPCfg>
<NTP>
    <Enabled> <!-- ro, req, xs:boolean, 是否启用 NTP --> </Enabled>
    <NTPServerIP> <!-- ro, req, xs:string, NTP 服务器 IP --> </NTPServerIP>
    <NTPServerPort> <!-- ro, req, xs:integer, NTP 服务器端口--> </NTPServerPort>
    <Interval> <!-- ro, req, xs:integer, 校时间隔 --> </Interval>
</NTP>
<ChannelCfg>
    <MaxAnalogChanNum> <!-- ro, req, xs:integer, 最大模拟通道个数--> </MaxAnalogChanNum>
    <AnalogChanNum> <!-- ro, req, xs:integer, 启用的模拟通道个数 --> </AnalogChanNum>
    <MaxDigitalChanNum> <!-- ro, req, xs:integer, 最大数字通道个数--> </MaxDigitalChanNum>
    <DigitalChanNum> <!-- ro, req, xs:integer, 启用的数字通道个数 --> </DigitalChanNum>
</ConfigSet>

```