

soundControl 接口说明

文档履历

版本号	日期	制/修订人	内容描述
V0.1	2016-6-12		初稿
V0.2	2016-8-9		更新接口 getFrameCount; 完善 SoundCtrl 和 SoundControlOpsS 结构体说明

目 录

soundControl 接口说明.....	1
1. 概述.....	1
1.1. 编写目的.....	1
1.2. 适用范围.....	1
1.3. 相关人员.....	1
2. 模块介绍.....	2
2.1. 功能介绍.....	2
2.2. 相关术语介绍.....	2
3. 接口函数.....	3
3.1. SoundDeviceDestroy.....	3
3.2. SoundDeviceSetFormat.....	3
3.3. SoundDeviceStart.....	3
3.4. SoundDeviceStop.....	3
3.5. SoundDevicePause.....	3
3.6. SoundDeviceWrite.....	3
3.7. SoundDeviceReset.....	4
3.8. SoundDeviceGetCachedTime.....	4
3.9. SoundDeviceGetFrameCount.....	4
3.10. SoundDeviceSetPlaybackRate.....	4
4. 数据结构.....	5
4.1. struct SoundCtrl.....	5
4.2. struct SoundControlOpsS.....	5
5. Declaration.....	7

1. 概述

1.1. 编写目的

设计应用层对音频数据输出接口，指导具体应用的开发、使用和后续维护。

1.2. 适用范围

Cdx2.6 以后版本。

1.3. 相关人员

2. 模块介绍

2.1. 功能介绍

2.2. 相关术语介绍

3. 接口函数

3.1. SoundDeviceDestroy

函数原型	void SoundDeviceDestroy(SoundCtrl* s)
功能	释放音频输出设备
参数	s: 音频输出设备句柄
返回值	
调用说明	

3.2. SoundDeviceSetFormat

函数原型	void SoundDeviceSetFormat(SoundCtrl* s, CdxPlaybkCfg* cfg)
功能	设置音频参数，采样率、通道数
参数	s: 音频输出设备句柄 Cfg: 音频信息结构体
返回值	
调用说明	

3.3. SoundDeviceStart

函数原型	int SoundDeviceStart(SoundCtrl* s)
功能	准备播放
参数	s: 音频输出设备句柄
返回值	成功: 0 失败: 返回错误码
调用说明	

3.4. SoundDeviceStop

函数原型	int SoundDeviceStop(SoundCtrl* s)
功能	停止播放
参数	s: 音频输出设备句柄
返回值	成功: 返回 0 失败: 返回错误码
调用说明	

3.5. SoundDevicePause

函数原型	int SoundDevicePause(SoundCtrl* s)
功能	暂停播放
参数	s: 音频输出设备句柄
返回值	成功: 返回 0 失败: -1
调用说明	

3.6. SoundDeviceWrite

函数原型	int SoundDeviceWrite(SoundCtrl* s, void* pData, int nDataSize)
功能	往音频设备写数据

参数	s: 音频输出设备句柄; pData: 音频数据起始地址 nDataSize: 音频数据长度
返回值	成功: 返回 0 失败: -1
调用说明	重采样可以在该接口中实现

3.7. SoundDeviceReset

函数原型	int SoundDeviceReset(SoundCtrl* s)
功能	重置播放
参数	s: 音频输出设备句柄;
返回值	成功: 返回 0 失败: -1
调用说明	

3.8. SoundDeviceGetCachedTime

函数原型	int SoundDeviceGetCachedTime(SoundCtrl* s)
功能	获取音频设备中缓存的音频数据
参数	s: 音频输出设备句柄;
返回值	成功: 返回缓冲数据时长 (ms) 失败: -1
调用说明	用于做音视频同步

3.9. SoundDeviceGetFrameCount

函数原型	int SoundDeviceGetFrameCount(SoundCtrl* s)
功能	获取音频设备中缓存的音频帧数 (用于 android 平台)
参数	s: 音频输出设备句柄;
返回值	成功: 返回缓冲音频帧数 失败: -1
调用说明	用于判断是否播放结束

3.10. SoundDeviceSetPlaybackRate

函数原型	int SoundDeviceSetPlaybackRate(SoundCtrl* s, const XAudioPlaybackRate*rate)
功能	设置速率播放
参数	s: 音频输出设备句柄; Rate: 播放速率
返回值	成功: 返回缓冲音频帧数 失败: -1
调用说明	Android6.0 之后才支持此功能

4. 数据结构

4.1. struct SoundCtrl

名称	struct SoundCtrl		
功能描述	字幕输出的结构体		
属性	类型	初始值	描述
ops	struct SoundControlOpsS*	NULL	音频输出的操作函数集合

4.2. struct SoundControlOpsS

名称	struct SoundControlOpsS	
功能描述	音频输出的接口定义，是其操作函数集合	
方法		
destory	原型	void (*destory)(SoundCtrl* s);
	参数	s: 句柄
	返回值	
	功能	关闭句柄，释放资源
setFormat	原型	void (*show)(SoundCtrl* s, unsigned int nSampleRate, unsigned int nChannelNum);
	参数	s: 音频输出设备句柄 nSampleRate: 采样率 nChannelNum: 通道数
	返回值	
	功能	设置音频采样率和通道数
start	原型	int (*start)(SoundCtrl* s);
	参数	s: 句柄
	返回值	成功: 返回 0 失败: 返回错误码
	功能	开始播放
stop	原型	int (*stop)(SoundCtrl* s);
	参数	s: 句柄
	返回值	成功: 返回 0 失败: 返回错误码
	功能	退出播放
pause	原型	int (*pause)(SoundCtrl* s);
	参数	s: 句柄
	返回值	成功: 返回 0 失败: 返回错误码
	功能	暂停播放
write	原型	int (*write)(SoundCtrl* s, void* pData, int nDataSize);
	参数	s: 句柄; pData: 音频数据起始地址

		nDataSize: 音频数据长度
	返回值	成功: 返回 0 失败: 返回错误码
	功能	往音频设备写数据
reset	原型	int (*reset) (SoundCtrl* s);
	参数	s: 句柄
	返回值	成功: 返回 0 失败: 返回错误码
	功能	重置播放
getCachedTime	原型	int (*getCachedTime) (SoundCtrl* s);
	参数	s: 句柄
	返回值	成功: 返回 0 失败: 返回错误码
	功能	获取音频设备缓冲数据
getFrameCount	原型	int (*getFrameCount) (SoundCtrl* s);
	参数	s: 句柄
	返回值	成功: 返回 0 失败: 返回错误码
	功能	获取音频设备剩余音频帧数
setPlaybackRate	原型	int (*setPlaybackRate) (SoundCtrl* s, const XAudioPlaybackRate *rate);
	参数	s: 句柄 Rate: 播放速率
	返回值	成功: 返回 0 失败: 返回错误码
	功能	获取音频设备剩余音频帧数

5. Declaration

This document is the original work and copyrighted property of Allwinner Technology (“Allwinner”). Reproduction in whole or in part must obtain the written approval of Allwinner and give clear acknowledgement to the copyright owner.

The information furnished by Allwinner is believed to be accurate and reliable. Allwinner reserves the right to make changes in circuit design and/or specifications at any time without notice. Allwinner does not assume any responsibility and liability for its use. Nor for any infringements of patents or other rights of the third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Allwinner. This datasheet neither states nor implies warranty of any kind, including fitness for any particular application.