

音频解码

文件标识：RK-SYS1-MPI-ADEC

发布版本：V0.2.0

日期：2021.1

文件密级：☐绝密 ☐秘密 ☐内部资料 ☒公开

免责声明

本文档按“现状”提供，瑞芯微电子股份有限公司（“本公司”，下同）不对本文档的任何陈述、信息和内容的准确性、可靠性、完整性、适销性、特定目的性和非侵权性提供任何明示或暗示的声明或保证。本文档仅作为使用指导的参考。

由于产品版本升级或其他原因，本文档将可能在未经任何通知的情况下，不定期进行更新或修改。

商标声明

“Rockchip”、“瑞芯微”、“瑞芯”均为本公司的注册商标，归本公司所有。

本文档可能提及的其他所有注册商标或商标，由其各自拥有者所有。

版权所有 © 2021 瑞芯微电子股份有限公司

超越合理使用范畴，非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

瑞芯微电子股份有限公司

Rockchip Electronics Co., Ltd.

地址：福建省福州市铜盘路软件园A区18号

网址：www.rock-chips.com

客户服务电话：+86-4007-700-590

客户服务传真：+86-591-83951833

客户服务邮箱：fae@rock-chips.com

前言

概述

本文档主要介绍ADEC的API和数据类型。

产品版本

芯片名称	内核版本
RK356X	4.19

读者对象

本文档（本指南）主要适用于以下工程师：

技术支持工程师

软件开发工程师

修订记录

版本号	作者	修改日期	修改说明
v0.1.0	周弟东	2021-1-4	初始版本
v0.2.0	周弟东	2021-1-22	修改描述错误

目录

音频解码

- 目录
- 基本概念
- 举例
- API 参考
 - RK_MPI_ADEC_CreateChn
 - RK_MPI_ADEC_DestroyChn
 - RK_MPI_ADEC_SendStream
 - RK_MPI_ADEC_GetFrame
 - RK_MPI_ADEC_ReleaseFrame
 - RK_MPI_ADEC_SendEndOfStream
 - RK_MPI_ADEC_QueryChnStat
- 数据类型
 - ADEC_CHN
 - ADEC_MAX_CHN_NUM
 - ADEC_MODE_E
 - ADEC_ATTR_CODEC_S
 - ADEC_CHN_ATTR_S
 - AUDIO_FRAME_INFO_S
 - ADEC_CHN_STATE_S
- 错误码
- 音频编解码协议说明

基本概念

ADEC MPI支持用户模式的方式：发送码流，获取解码帧。也支持通过SYS模块的Bind接口，将一个ADEC通道Bind到AO通道，实现解码播放功能。

举例

```
RK_S32 s32ret;
ADEC_CHN_ATTR_S stAdecAttr;
AUDIO_STREAM_S stAudioStream;
AUDIO_FRAME_INFO_S *pstFrmInfo = RK_NULL;
AdecContext *params = RK_NULL;

// init adec params
```

```

ADEC_CHN AdChn = 0;
RK_BOOL  bBlock = RK_TRUE;
stAdecAttr.enType = RK_AUDIO_ID_MP3;
stAdecAttr.enMode = ADEC_MODE_STREAM;
stAdecAttr.stAdecCodec.u32Channels = 2;
stAdecAttr.stAdecCodec.u32SampleRate = 44100;
stAdecAttr.stAdecCodec.u32BitPerCodedSample = 0;
stAdecAttr.extraDataSize = 0;
stAdecAttr.extraData = RK_NULL;

// create a adec achn
s32ret = RK_MPI_ADEC_CreateChn(AdChn, &stAdecAttr);
if (s32ret) {
    RT_LOGE("create adec chn %d err:0x%x\n", AdChn, s32ret);
    return s32ret;
}

s32ret = RK_MPI_ADEC_GetFrame(AdChn, pstFrmInfo, bBlock);
if (s32ret == RK_SUCCESS) {
    MB_BLK bBlk = pstFrmInfo->pstFrame->pMbBlk;
    RK_VOID *pstFrame = RK_MPI_MB_Handle2VirAddr(bBlk);
    RK_S32 frameSize = pstFrmInfo->pstFrame->u32Len;
    RK_S32 seq = pstFrmInfo->pstFrame->u32Seq;
    if (pstFrame) {
        // release frame
        RK_MPI_ADEC_ReleaseFrame(AdChn, pstFrmInfo);
    }
}

// destroy a adec chn
s32ret = RK_MPI_ADEC_DestroyChn(AdChn);
if (RK_SUCCESS != s32ret) {
    return s32ret;
}

```

详细测试DEMO，请参考发布文件：test_mpi_adec.cpp。

API 参考

该功能模块为用户提供以下API:

- [RK MPI ADEC CreateChn](#): 创建音频解码通道。
- [RK MPI ADEC DestroyChn](#): 销毁音频解码通道。
- [RK MPI ADEC SendStream](#): 发送音频码流到音频解码通道。
- [RK MPI ADEC ClearChnBuf](#): 清除ADEC通道中当前的音频数据缓存。
- [RK MPI ADEC GetFrame](#): 获取音频解码帧数据。
- [RK MPI ADEC ReleaseFrame](#): 释放音频解码帧数据。
- [RK MPI ADEC SendEndOfStream](#): 向解码器发送码流结束标识符。
- [RK MPI ADEC QueryChnStat](#): 查询ADEC通道中当前的音频数据缓存状态。

RK_MPI_ADEC_CreateChn

【描述】

创建音频解码通道。

【语法】

RK_S32 RK_MPI_ADEC_CreateChn ([ADEC_CHN](#) AdChn, const [ADEC_CHN_ATTR_S](#) *pstAttr);

【参数】

参数名	描述	输入/输出
AdChn	通道号。 取值范围：[0, ADEC_MAX_CHN_NUM)。	输入
pstAttr	通道属性指针。	输入

【返回值】

返回值	描述
0	成功。
非0	失败，请参见 错误码 。

【注意】

- 音频解码支持的解码协议：RK_AUDIO_ID_MP3、RK_AUDIO_ID_AAC、RK_AUDIO_ID_ADPCM_G722、RK_AUDIO_ID_ADPCM_G726等，参见rk_common.h中RK_CODEC_ID_E枚举音频定义。
- g726音频格式需要设置码流码字（codecwords），支持的协议说明见表[音频编解码协议说明](#)。
- 音频解码的初始化属性必须设置码流的采样率（u32SampleRate）、声道数（u32Channels）、codec id（enType）三个参数。解码模式默认为PACK模式，支持STREAM模式，但是建议使用PACK模式。
- 在通道闲置时才能使用此接口，如果通道已经被创建，则返回通道已经创建的错误。

RK_MPI_ADEC_DestroyChn

【描述】

销毁音频解码通道。

【语法】

RK_S32 RK_MPI_ADEC_DestroyChn([ADEC_CHN](#) AdChn);

【参数】

参数名	描述	输入/输出
AdChn	通道号。 取值范围：[0, ADEC_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非0	失败，请参见 错误码 。

【注意】

- 通道未创建的情况下调用此接口会返回RK_ERR_ADEC_UNEXIST。
- 建议通道使用完成后调用此接口。

RK_MPI_ADEC_SendStream

【描述】
向音频解码通道发送码流。

【语法】

```
RK_S32 RK_MPI_ADEC_SendStream (ADEC_CHN AdChn, const AUDIO_STREAM_S *pstStream, RK_BOOL bBlock);
```

【参数】

参数名	描述	输入/输出
AdChn	通道号。 取值范围：[0,ADEC_MAX_CHN_NUM)。	输入
pstStream	音频码流。	输入
bBlock	阻塞标识。RK_TRUE：阻塞。RK_FALSE：非阻塞。	输入

【返回值】

返回值	描述
0	成功。
非0	失败，请参见错误码。

【注意】

- 创建解码通道时可以指定解码方式为pack方式或者stream方式：
 1. pack方式用于确定码流包为一帧的情况下，比如从网络直接获取的码流，从文件读取确切知道一帧边界的码流，效率较高。
 2. stream方式用于不确定码流包为一帧的情况下，效率较低，且可能会有延迟。
- 发送数据时必须保证通道已经被创建，否则直接返回失败。
- 支持阻塞或者非阻塞方式发送码流。
- 当阻塞方式发送码流时，如果用于缓存解码后的音频帧的buffer满，则此接口调用会被阻塞，直到解码后的音频帧数据被取走，或ADEC通道被销毁，建议选择阻塞方式发送码流。
- 确保发送给ADEC通道的码流数据的正确性，否则可能引起解码器异常退出。
- 音频申请内存的方式建议尽量使用malloc方式申请。

RK_MPI_ADEC_GetFrame

【描述】
获取音频解码帧数据。

【语法】

```
RK_S32 RK_MPI_ADEC_GetFrame (ADEC_CHN AdChn, AUDIO_FRAME_INFO_S *pstFrmlInfo, RK_BOOL bBlock);
```

【参数】

参数名	描述	输入/输出
AdChn	通道号。 取值范围：[0, ADEC_MAX_CHN_NUM)。	输入
pstFrmInfo	音频帧数据结构体。	输出
bBlock	是否以阻塞方式获取	输入

【返回值】

返回值	描述
0	成功。
非0	失败，请参见错误码。

【注意】

- 必须在ADEC通道创建之后调用。
- 使用本接口获取音频帧数据时，建议发送码流时按pack方式（帧模式）发送。
- 使用本接口获取音频帧数据时，如果发送码流按stream发送，请务必保证获取解码帧数据的及时性；如果sendStream和getFram接口都是阻塞的，需要各自在不同的线程，否则会阻塞住。
- getFream成功后，需要调用ReleaseFrame。
- 使用本接口获取音频数据时，ADEC BIND AO的方式不需要用此接口，请先解除ADEC与AO的绑定关系，否则获取到的帧是不连续的。

RK_MPI_ADEC_ReleaseFrame

【描述】

释放获取到的音频解码帧数据。

【语法】

RK_S32 RK_MPI_ADEC_ReleaseFrame ([ADEC_CHN](#) AdChn, [AUDIO_FRAME_INFO_S](#) *pstFrmInfo);

【参数】

参数名	描述	输入/输出
AdChn	通道号。 取值范围：[0, ADEC_MAX_CHN_NUM)。	输入
pstFrmInfo	音频帧数据结构。	输入

【返回值】

返回值	描述
0	成功。
非0	失败，请参见错误码。

【注意】

- 必须在ADEC通道创建之后调用。
- 本接口必须与接口RK_MPI_ADEC_GetFrame配合使用。

RK_MPI_ADEC_SendEndOfStream

【描述】

向解码器发送码流结束标识符。

【语法】

RK_S32 RK_MPI_ADEC_SendEndOfStream ([ADEC_CHN](#) AdChn, RK_BOOL bInstant);

【参数】

参数名	描述	输入/输出
AdChn	通道号。 取值范围：[0, ADEC_MAX_CHN_NUM)。	输入
bInstant	是否立刻清除解码器内部的缓存数据。 取值范围：RK_FALSE: 不清除缓存数据，解码继续进行，直到解码完音频码流。 RK_TRUE: 立刻清除解码器内部的缓存数据。	输入

【返回值】

返回值	描述
0	成功。
非0	失败，请参见错误码。

【注意】

- 建议bInstant使用RK_FALSE。

RK_MPI_ADEC_QueryChnStat

【描述】

查询ADEC通道中当前的音频数据缓存状态。

【语法】

RK_S32 RK_MPI_ADEC_QueryChnStat ([ADEC_CHN](#) AdChn, [ADEC_CHN_STATE_S](#) *pstBufferStatus);

【参数】

参数名	描述	输入/输出
AdChn	通道号。 取值范围：[0, ADEC_MAX_CHN_NUM)。	输入
pstBufferStatus	缓存状态结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非0	失败，请参见错误码。

【注意】

- 在ADEC通道成功启用后再调用此接口。

数据类型

ADEC_CHN

【说明】

定义 ADEC 通道。

【定义】

```
typedef RK_S32 ADEC_CHN;
```

ADEC_MAX_CHN_NUM

【说明】

定义音频解码通道的最大个数。

【定义】

```
#define ADEC_MAX_CHN_NUM 32
```

ADEC_MODE_E

【说明】

定义解码方式。

【定义】

```
typedef enum rkADEC_MODE_E {  
    /*  
     * require input is valid dec pack(a complete frame encode result),  
     * e.g.the stream get from AENC is a valid dec pack, the stream know  
     * actually pack len from file is also a dec pack.  
     * this mode is high-performative*/  
    ADEC_MODE_PACK = 0,  
    /*  
     * input is stream,low-performative, if you couldn't find out whether  
     * a stream is vaild dec pack,you could use this mode  
     */  
    ADEC_MODE_STREAM,  
    ADEC_MODE_BUTT  
} ADEC_MODE_E;
```

【成员】

成员名称	描述
ADEC_MODE_PACK	pack方式解码。
ADEC_MODE_STREAM	stream方式解码。

【注意事项】

- pack方式用于用户确认当前码流包为一帧数据编码结果的情况下，解码器会直接进行对其解码，如果不是一帧，解码器会出错。这种模式的效率比较高，在使用AENC模块编码的码流包如果没有破坏，均可以使用此方式解码。
- stream方式用于用户不能确认当前码流包是不是一帧数据的情况下，解码器需要对码流进行判断并缓存，此工作方式的效率低下，一般用于读文件码流送解码或者不确定码流包边界的情况。当然由于语音编码码流长度固定，很容易确定在码流中的帧边界，推荐使用pack方式解码。

ADEC_ATTR_CODEC_S

【说明】

定义音频解码器属性结构体

【定义】

```
typedef struct rkaDEC_ATTR_CODEC_S {
    RK_U32  u32Channels;
    RK_U32  u32SampleRate;
} ADEC_ATTR_CODEC_S;
```

【成员】

成员名称	描述
u32Channels	码流声道。
u32SampleRate	码流采样率。

ADEC_CHN_ATTR_S

【说明】

定义解码通道属性结构体。

【定义】

```
typedef struct rkaDEC_CH_ATTR_S {
    RK_CODEC_ID_E    enType;
    ADEC_MODE_E      enMode;      /*decode mode*/
    ADEC_ATTR_CODEC_S stAdecCodec;
    RK_U32           u32BufCount;
    MB_BLK           extraData;
    RK_U32           extraDataSize;
} ADEC_CHN_ATTR_S;
```

【成员】

成员名称	描述
enType	解码器codec id。
enMode	解码方式。
stAdecCodec	解码器属性参数。
u32BufCount	音频解码缓存个数。
extraData	解码器外部数据。
extraDataSize	解码器外部数据大小。

AUDIO_FRAME_INFO_S

【说明】
定义解码后的音频帧信息结构体。
【定义】

```
typedef struct rkAUDIO_FRAME_INFO_S {
    AUDIO_FRAME_S *pstFrame; /*frame ptr*/
    RK_U32      u32Id;    /*frame id*/
} AUDIO_FRAME_INFO_S;
```

【成员】

成员名称	描述
pstFrame	音频帧指针。
u32Id	音频帧的索引。

ADEC_CHN_STATE_S

【说明】
定义音频解码通道的数据缓存状态结构体。
【定义】

```
typedef struct rkADEC_CH_STATE_S {
    RK_BOOL bEndOfStream;    /* EOS flag */
    RK_U32  u32BufferFrmNum; /* total number of channel buffer */
    RK_U32  u32BufferFreeNum; /* free number of channel buffer */
    RK_U32  u32BufferBusyNum; /* busy number of channel buffer */
} ADEC_CHN_STATE_S;
```

【成员】

成员名称	描述
bEndOfStream	解码码流结束状态。
u32BufferFrmNum	解码通道总的缓存块数。
u32BufferFreeNum	可用的空闲缓存块数。
u32BufferBusyNum	被占用缓存块数。

错误码

音频解码API错误码如下所示。

错误代码	宏定义	描述
0xA00D8001	RK_ERR_ADEC_INVALID_DEVID	音频输出设备号无效
0xA00D8002	RK_ERR_ADEC_INVALID_CHNID	音频输出通道号无效
0xA00D8003	RK_ERR_ADEC_ILLEGAL_PARAM	音频输出参数设置无效
0xA00D8004	RK_ERR_ADEC_EXIST	音频输出设备或通道没使能
0xA00D8005	RK_ERR_ADEC_UNEXIST	输出空指针错误
0xA00D8006	RK_ERR_ADEC_NULL_PTR	音频输出设备属性未设置
0xA00D8007	RK_ERR_ADEC_NOT_CONFIG	操作不被支持
0xA00D8008	RK_ERR_ADEC_NOT_SUPPORT	操作不允许
0xA00D8009	RK_ERR_ADEC_NOT_PERM	系统内存不足
0xA00D800C	RK_ERR_ADEC_NOMEM	音频输出缓存不足
0xA00D800D	RK_ERR_ADEC_NOBUF	音频输出缓存为空
0xA00D800E	RK_ERR_ADEC_BUF_EMPTY	音频输出缓存为满
0xA00D8010	RK_ERR_ADEC_BUF_FULL	音频输出缓存为满
0xA00D8010	RK_ERR_ADEC_SYS_NOTREADY	音频输出系统未初始化
0xA00D8040	RK_ERR_ADEC_DECODER_ERR	音频输出系统忙
0xA00D8041	RK_ERR_ADEC_BUF_LACK	AO VQE 处理错误

音频编解码协议说明

协议	采样率	码率 (kbps)	码字 (bits)
g726	8kHz	40	5
		32	4
		24	3
		16	2