

中科院声学所离线识别引擎

SOX接口设计

版 本 历 史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本/状态 | 作者 | 参与者 | 起止日期 | 备注 |
| V1.0 | 王超锋 |  | 2021-11-17 | 创建文件 |
|  |  |  |  |  |

目录

[1.概述 3](#_Toc2946)

[1.1 编写目的 3](#_Toc31519)

[2.总体设计 4](#_Toc12306)

[2.1 总体需求 4](#_Toc17097)

[3.SOX接口设计 5](#_Toc32546)

[3.1 通用 API 6](#_Toc4363)

[4.代码目录 8](#_Toc22819)

[4.1 代码目录结构 8](#_Toc20311)

[4.2 主要代码模块说明 17](#_Toc26997)

[4.2.1 转码模块 17](#_Toc11354)

[4.2.2 控制模块 17](#_Toc22944)

# 1.概述

1

**1.1 编写目的**

本文档主要描述sox内部接口设计及接口中参数的数据结构，本文档的读者对象为离线识别系统的开发人员、测试人员、系统维护人员及接入识别系统的第三方业务人员，通过本文档能够从总体上了解识别系统内部的数据流向及处理方式。

本说明给出SOX的接口设计说明，包括接口安全、版本兼容性、数据格式、服务端的异常处理等。

目的在于：

* 为开发人员提供依据；
* 为代码修改、维护提供条件；

# 2.总体设计

2

## 2.1 总体需求

接口需求：

1，接口参数简单明了，参数有实际意义；

2，接口功能定义明确；

3，接口的数据特性明确；

4，接口的规格和技术要求明确；

5，接口程序的数据处理逻辑清晰；

# 3.SOX接口设计

3

本节主要介绍soxAPI使用。

SoX 处理音频的基本流程如下：



## 3.1 通用 API

1.sox\_init()：

在使用SOX库之前，必须初始化整个库的全局参数。函数返回一个整数，如果返回的是SOX\_SUCCESS这个枚举值，代码初始化成功；

2.sox\_init()：

退出SOX在。

3.sox\_open\_read()：

初始化输入文件。

4.sox\_open\_write()：

初始化输出文件。

5.sox\_create\_chain()：

构造效果器链。

1. sox\_create\_cffect()

构建一个输入数据的效果器。

1. sox\_effect\_options()

输入文件配置到这个效果器。

1. sox\_add\_effect()

效果器增加到效果器链。

1. sox\_flow\_effects()

运行效果器链。

# 4.代码目录

4

## 4.1 代码目录结构

sox-14.4.2

├── 8svx.c

├── adpcm.c

├── adpcm.h

├── adpcms.c

├── adpcms.h

├── aifc-fmt.c

├── aiff.c

├── aiff-fmt.c

├── aiff.h

├── al-fmt.c

├── alsa.c

├── amr.h

├── amr-nb.c

├── amr-wb.c

├── ao.c

├── au.c

├── avr.c

├── band.h

├── bend.c

├── biquad.c

├── biquad.h

├── biquads.c

├── caf.c

├── cdr.c

├── chorus.c

├── CMakeLists.txt

├── compand.c

├── compandt.c

├── compandt.h

├── contrast.c

├── coreaudio.c

├── cvsd.c

├── cvsdfilt.h

├── cvsd-fmt.c

├── cvsd.h

├── dat.c

├── dcshift.c

├── delay.c

├── dft\_filter.c

├── dft\_filter.h

├── dither.c

├── dither.h

├── divide.c

├── downsample.c

├── dvms-fmt.c

├── earwax.c

├── echo.c

├── echos.c

├── effects.c

├── effects.h

├── effects\_i.c

├── effects\_i\_dsp.c

├── example0.c

├── example1.c

├── example2.c

├── example3.c

├── example4.c

├── example5.c

├── example6.c

├── f4-fmt.c

├── f8-fmt.c

├── fade.c

├── fap.c

├── fft4g.c

├── fft4g.h

├── fifo.h

├── fir.c

├── firfit.c

├── flac.c

├── flanger.c

├── formats.c

├── formats.h

├── formats\_i.c

├── g711.c

├── g711.h

├── g721.c

├── g723\_24.c

├── g723\_40.c

├── g72x.c

├── g72x.h

├── gain.c

├── getopt.c

├── gsm.c

├── gsrt.c

├── hcom.c

├── hilbert.c

├── htk.c

├── ignore-warning.h

├── ima-fmt.c

├── ima\_rw.c

├── ima\_rw.h

├── input.c

├── ladspa.c

├── ladspa.h

├── la-fmt.c

├── libsox.c

├── libsox\_i.c

├── loudness.c

├── lpc10.c

├── lu-fmt.c

├── Makefile.am

├── Makefile.in

├── mat4.c

├── mat5.c

├── maud.c

├── mcompand.c

├── mcompand\_xover.h

├── monkey.wav

├── mp3.c

├── mp3-util.h

├── noiseprof.c

├── noisered.c

├── noisered.h

├── nulfile.c

├── optional-fmts.am

├── opus.c

├── oss.c

├── output.c

├── overdrive.c

├── pad.c

├── paf.c

├── phaser.c

├── prc.c

├── pulseaudio.c

├── pvf.c

├── rate.c

├── rate\_filters.h

├── rate\_half\_fir.h

├── rate\_poly\_fir0.h

├── rate\_poly\_fir.h

├── raw.c

├── raw-fmt.c

├── raw.h

├── remix.c

├── repeat.c

├── reverb.c

├── reverse.c

├── s1-fmt.c

├── s2-fmt.c

├── s3-fmt.c

├── s4-fmt.c

├── sd2.c

├── sf.c

├── silence.c

├── sinc.c

├── skeleff.c

├── skelform.c

├── smp.c

├── sndfile.c

├── sndio.c

├── sounder.c

├── soundtool.c

├── sox.c

├── soxconfig.h.cmake

├── soxconfig.h.in

├── sox-fmt.c

├── sox.h

├── sox\_i.h

├── soxomp.h

├── sox\_sample\_test.c

├── sox\_sample\_test.h

├── spectrogram.c

├── speed.c

├── sphere.c

├── splice.c

├── stat.c

├── stats.c

├── stretch.c

├── sunaudio.c

├── swap.c

├── synth.c

├── tempo.c

├── testall.bat

├── testall.sh

├── test-comments

├── tests.bat

├── tests.sh

├── tremolo.c

├── trim.c

├── tx16w.c

├── u1-fmt.c

├── u2-fmt.c

├── u3-fmt.c

├── u4-fmt.c

├── ul-fmt.c

├── upsample.c

├── util.c

├── util.h

├── vad.c

├── voc.c

├── vol.c

├── vorbis.c

├── vox.c

├── vox-fmt.c

├── vox.h

├── w64.c

├── wav.c

├── waveaudio.c

├── wavpack.c

├── win32-glob.c

├── win32-glob.h

├── win32-ltdl.c

├── win32-ltdl.h

├── wve.c

├── xa.c

├── xi.c

├── xmalloc.c

└── xmalloc.h

## 4.2 主要代码模块说明

### 4.2.1 转码模块

g72x.c该文件主要实现g72x格式文件转换。

mp3.c该文件主要实现mp3格式文件转换。

### 4.2.2 控制模块

sox.c该文件主要实现流程处理函数的实现。