



AX GZIPD API 文档

文档版本: V1.1

发布日期: 2024/01/31

目 录

前 言	4
修订历史	5
1 概述	6
1.1 概述	6
1.2 功能描述	6
1.2.1 压缩工具	6
1.2.2 解压功能	6
1.2.3 API 调用流程	7
2 API 参考	8
2.1 AX_GZIPD_Init	8
2.2 AX_GZIPD_DelInit	10
2.3 AX_GZIPD_CreateHandle	11
2.4 AX_GZIPD_DestroyHandle	13
2.5 AX_GZIPD_Config	14
2.6 AX_GZIPD_TilesRun	15
2.7 AX_GZIPD_LastTileRun	17
2.8 AX_GZIPD_WaitFinish	19
3 数据结构	20
AX_GZIPD_HEADER_INFO_T	20
AX_GZIPD_BUF_INFO_T	21
AX_GZIPD_IO_PARAM_T	22
AX_GZIPD_RESULT_INFO_T	23

AX_GZIPD_COMPLETE_STS_E	24
4 错误码.....	25
5 日志.....	26

权利声明

爱芯元智半导体股份有限公司或其许可人保留一切权利。

非经权利人书面许可，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。

注意

您购买的产品、服务或特性等应受商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非商业合同另有约定，本公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

前 言

本文档主要介绍 AX GZIPD 的 API 接口说明。

适用产品

AX620E 系列产品（AX630C、AX620Q）

适读人群

- 终端用户
- 售前
- 售后
- 技术人员

符号与格式定义

符号/格式	说明
xxx	表示您可以执行的命令行。
斜体	表示变量。如，“ <i>安装目录/AX620E_SDK_Vx.x.x/build</i> ”中的“ <i>安装目录</i> ”是一个变量，由您的实际环境决定。
☞ 说明/备注：	表示您在使用产品的过程中，我们向您说明的事项。
！ 注意：	表示您在使用产品的过程中，需要您特别注意的事项。

修订历史

文档版本	发布时间	修订说明
V1.0	2023/09/19	Initial
V1.1	2024/01/31	更新数据结构部分

1 概述

本章节将对 AX GZIPD 模块做简要介绍，包含模块涉及的重要概念。

1.1 概述

AX GZIPD (AXERA GZIP Decompress) API 是本司提供的可解压自定义的，类似 GZIP 格式的压缩数据的应用 API，用户基于此 API 可以开发出相关解压功能应用，当前 AX GZIPD 支持的功能主要是快速解压以 AXERA 自定义格式定义的压缩文件。

1.2 功能描述

1.2.1 压缩工具

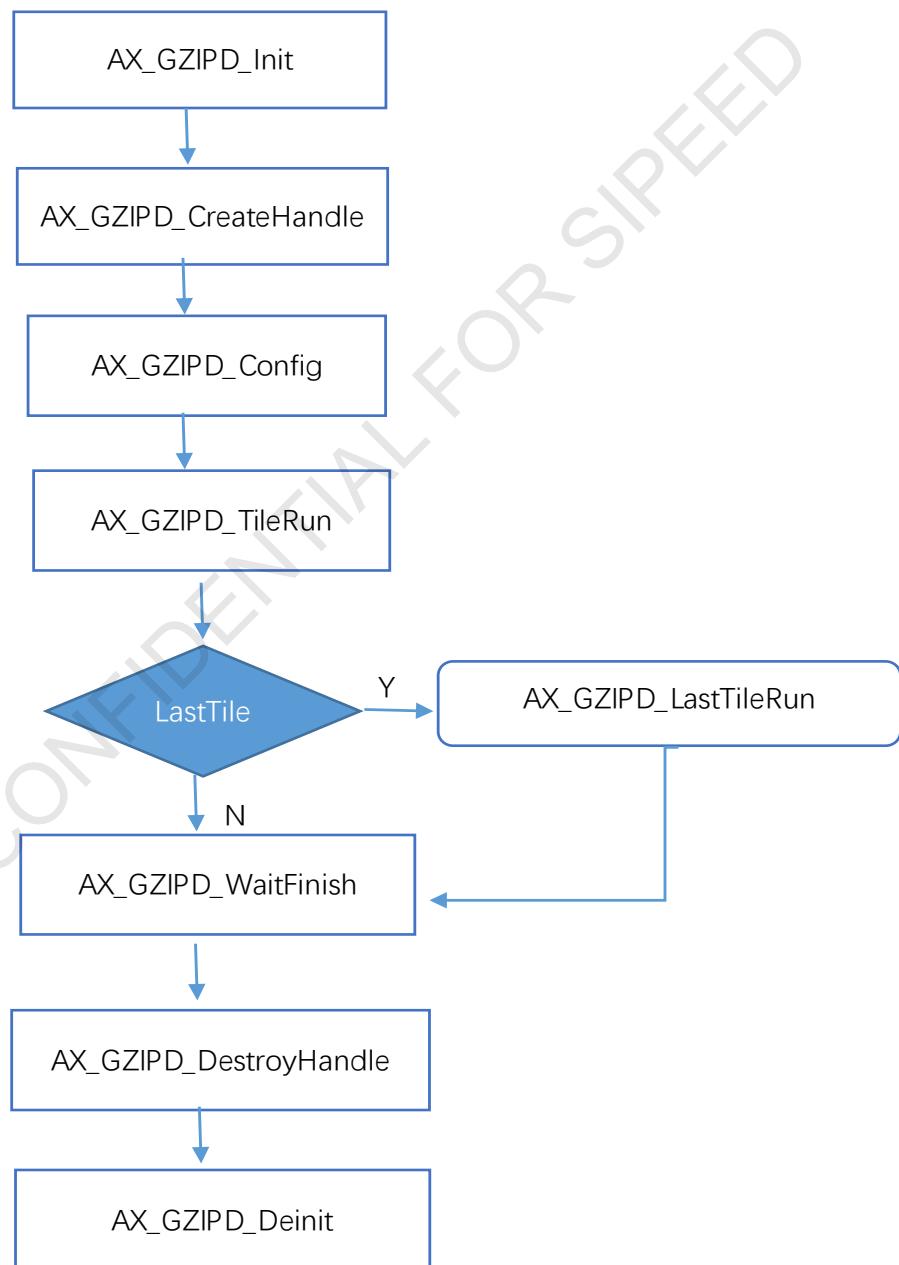
使用 AXERA 专用压缩工具 ax_gzip 来压缩原始数据，生成对应的压缩文件；

压缩工具 ax_gzip 在 SDK 包的 tools 目录下。

1.2.2 解压功能

将全部压缩数据按照 tile size 大小分割为若干个 tile，依次把 tile 信息配置到 gzipd 里，启动解压过程，继续把后续 tile 的信息设置到 gzipd 里，直到全部压缩数据被 gzipd 解压完成。

1.2.3 API 调用流程



2 API 参考

2.1 AX_GZIPD_Init

【描述】

对 GZIPD 模块初始化。

【语法】

```
AX_S32 AX_GZIPD_Init(AX_VOID)
```

【参数】

无

【返回值】

返回值	描述
0	成功
非 0	失败, 返回错误码

【需求】

- 头文件: ax_gzipd_api.h
- 库文件: libax_gzipd.so

【注意】

- 本接口支持重复调用。
- 本接口和 AX_GZIPD_DeInit 一一对应，

【举例】

无

【相关主题】

函数: [AX_GZIPD_DeInit](#)

AEXRA CONFIDENTIAL FOR SIPEED

2.2 AX_GZIPD_DeInit

【描述】

对 GZIPD 模块去初始化。

【语法】

```
AX_S32 AX_GZIPD_DeInit(AX_VOID)
```

【参数】

无

【返回值】

返回值	描述
0	成功
非 0	失败, 返回错误码

【需求】

- 头文件: ax_gzipd_api.h
- 库文件: libax_gzipd.so

【注意】

无

【举例】

无

【相关主题】

函数: [AX_GZIPD_Init](#)

2.3 AX_GZIPD_CreateHandle

【描述】

创建 MD 通道

【语法】

```
AX_S32 AX_GZIPD_CreateHandle(AX_U32 *pHandle, AX_VOID *pGZIPData,  
AX_GZIPD_HEADER_INFO_T *pOutHeaderInfo);
```

【参数】

参数名称	描述	输入/输出
pHandle	对应一个解压任务的索引值	输出
pGZIPData	指向 AXGZIP 格式的压缩数据	输入
pOutHeaderInfo	描述压缩数据头信息	输出

【返回值】

返回值	描述
0	成功
非 0	失败, 返回错误码

【需求】

- 头文件: ax_gzipd_api.h
- 库文件: libax_gzipd.so

【注意】

无

【举例】

无

【相关主题】

函数: [AX_GZIPD_DestroyHandle](#)

AEXRA CONFIDENTIAL FOR SIPEED

2.4 AX_GZIPD_DestroyHandle

【描述】

销毁之前创建的 handle

【语法】

```
AX_S32 AX_GZIPD_DestroyHandle(AX_U32 handle);
```

【参数】

参数名称	描述	输入/输出
handle	一个解压任务的索引值	输入

【返回值】

返回值	描述
0	成功
非 0	失败, 返回错误码

【需求】

- 头文件: ax_gzipd_api.h
- 库文件: libax_gzipd.so

【注意】

无

【举例】

无

【相关主题】

函数: [AX_GZIPD_CreateHandle](#)

2.5 AX_GZIPD_Config

【描述】

配置 GZIPD 模块的工作参数

【语法】

```
AX_S32 AX_GZIPD_Config (AX_U32 handle, AX_GZIPD_IO_PARAM_T  
*pGzipdInputParam);
```

【参数】

参数名称	描述	输入/输出
handle	一个解压任务的索引值	输入
pGzipdInputParam	GZIPD 的工作参数	输入

【返回值】

返回值	描述
0	成功
非 0	失败, 返回错误码

【需求】

- 头文件: ax_gzipd_api.h
- 库文件: libax_gzipd.so

【注意】

无

【举例】

无

【相关主题】

无

2.6 AX_GZIPD_TilesRun

【描述】

执行解压

【语法】

```
AX_U32 AX_GZIPD_TilesRun(AX_U32 handle, AX_U64 u64TilesStartPhyaddr, AX_U64  
u64TilesDataBytesLen, AX_U32 *u32QueuedTilesNum);
```

【参数】

参数名称	描述	输入/输出
handle	一个解压任务的索引值	输入
u64TilesStartPhyaddr	指向待解压的压缩数据的首地址	输入
u64TilesDataBytesLen	待解压的压缩数据的数量（单位：字节）	输入
u32QueuedTilesNum	返回已解压数据的数量	输出

【返回值】

返回值	描述
0	成功
非 0	失败，返回错误码

【需求】

- 头文件: ax_gzipd_api.h
- 库文件: libax_gzipd.so

【注意】

无

【举例】

无

【相关主题】

函数: [AX_GZIPD_LastTileRun](#)

AEXRA CONFIDENTIAL FOR SPEED

2.7 AX_GZIPD_LastTileRun

【描述】

解压最后一个 Tile 的压缩数据

【语法】

```
AX_S32 AX_GZIPD_LastTileRun(AX_U32 handle, AX_U64 u64LastTilePhyAddr, AX_U32 u32LastTileSize);
```

【参数】

参数名称	描述	输入/输出
handle	一个解压任务的索引值	输入
u64LastTilePhyAddr	指向最后一个 Tile 大小压缩数据的首地址	输入
u32LastTileSize	最后一个 Tile 的大小值	输入

【返回值】

返回值	描述
0	成功
非 0	失败, 返回错误码

【需求】

- 头文件: ax_gzipd_api.h
- 库文件: libax_gzipd.so

【注意】

无

【举例】

无

【相关主题】

函数: [AX_GZIPD_TileRun](#)

AEXRA CONFIDENTIAL FOR SPEED

2.8 AX_GZIPD_WaitFinish

【描述】

等待解压任务执行完成

【语法】

```
AX_S32 AX_GZIPD_WaitFinish(AX_U32 handle, AX_GZIPD_RESULT_INFO_T *pResult);
```

【参数】

参数名称	描述	输入/输出
handle	一个解压任务的索引值	输入
pResult	解压任务结果	输出

【返回值】

返回值	描述
0	成功
非 0	失败, 返回错误码

【需求】

- 头文件: ax_gzipd_api.h
- 库文件: libax_gzipd.so

【注意】

无

【举例】

无

【相关主题】

无

3 数据结构

AX_GIZPD_HEADER_INFO_T

【说明】

AX-GZIP 压缩格式的文件头信息

【定义】

```
typedef struct {  
    AX_U8    cType[2];  
  
    AX_U16   u16BlkNum;  
  
    AX_U32   u32OutSize;  
  
    AX_U32   u32InSize;  
  
    AX_U32   u32Crc32;  
} AX_GZIPD_HEADER_INFO_T;
```

【成员】

成员名称	描述
cType[2]	AX-GZIP 压缩格式的 Magic Value
u16BlkNum	压缩数据的 Block 信息
u32OutSize	解压数据的大小值
u32InSize	压缩数据的大小值
u32Crc32	压缩数据的 CRC32 校验值

【注意】

无

【相关数据类型及接口】

无

AX_GZIPD_BUF_INFO_T

【说明】

数据占用内存

【定义】

```
typedef struct {  
  
    AX_U64 pPhyAddr;  
  
    AX_VOID *pVirAddr;  
  
    AX_U32 u32BufSize;  
} AX_GZIPD_BUF_INFO_T;
```

【成员】

成员名称	描述
pPhyAddr	数据内存的物理地址
pVirAddr	数据内存的虚拟地址
u32BufSize	数据占用内存的大小值

【注意】

无

【相关数据类型及接口】

无

AX_GZIPD_IO_PARAM_T

【说明】

压缩数据信息

【定义】

```
typedef struct {  
    AX_GZIPD_BUF_INFO_T stInBuf;  
  
    AX_GZIPD_BUF_INFO_T stOutBuf;  
  
    AX_U32 u32TileSize;  
  
    AX_GZIPD_HEADER_INFO_T headerInfo;  
  
    AX_U64 tilesNum;  
  
    AX_U64 lastTileLen;  
} AX_GZIPD_IO_PARAM_T;
```

【成员】

成员名称	描述
stInBuf	压缩数据的内存地址
stOutBuf	解压数据的内存地址
u32TileSize	一个 Tile 数据的大小值
headerInfo	压缩数据的文件头信息
tilesNum	把压缩数据分割成 tilesNum 个 u32TileSize 大小的数据断
lastTileLen	最后一个 Tile 数据段的大小值

【注意】

无

【相关数据类型及接口】

无

AX_GZIPD_RESULT_INFO_T

【说明】

解压完成信息

【定义】

```
typedef struct {  
    AX_U32 u32Handle;  
  
    GZIPD_INST_COMPLETE_STS_E u32Result;  
} AX_GZIPD_RESULT_INFO_T;
```

【成员】

成员名称	描述
u32Handle	一个解压任务的句柄
u32Result	解压完成状态

【注意】

无

【相关数据类型及接口】

无

AX_GZIPD_COMPLETE_STS_E

【说明】

解压任务完成状态

【定义】

```
typedef enum {  
  
    AX_GZIPD_COMPLETE_SUCCESS,  
  
    AX_GZIPD_COMPLETE_FAIL,  
  
    AX_GZIPD_COMPLETE_MAX,  
} AX_GZIPD_COMPLETE_STS_E
```

【成员】

成员名称	描述
AX_GZIPD_COMPLETE_SUCCESS	解压任务成功完成
AX_GZIPD_COMPLETE_FAIL	解压任务失败
AX_GZIPD_COMPLETE_MAX	状态最大值

【注意】

无

【相关数据类型及接口】

无

4 错误码

错误码详见《55 - AX 软件错误码文档》文档。

5 日志

➤ 设置 GZIPD 日志等级

```
echo [id] [level] > /proc/ax_proc/logctl
```

id: [0-45]表示单模块 id, 如果设置所有模块则用 all 表示;

level: log 等级, 0-7 数字越小代表 Log 等级越高;

示例: 设置 GZIPD 模块 Log 等级为 4

```
echo 45 4 > /proc/ax_proc/logctl
```

➤ 设置日志目的地

Log 输出到文件:

```
echo target file > /proc/ax_proc/logctl
```

Log 输出到终端:

```
echo target console > /proc/ax_proc/logctl, Log 输出到终端, 不再保存到文件
```

关闭 Log:

```
echo target null > /proc/ax_proc/logctl
```