
USB3.0 SC-SDK

用户指南

文档版本 V2.0.0

发布日期：2025-03-01

文档版本 V2.0.0-2025-03-01

修订记录

修订记录累积了每次文档更新的说明。最新版本的文档包含以前所有文档版本的更新内容。

Revision sheet 变更单			
Revision No. 版本号	Date 日期	Revision 变更内容	Sig. 签名
V2.0.0	20250301	整合原有 SDK 功能发布全新版本	11215wuc

目 录

1. 图像&通信(API).....	4
1.1 API 接口	4
1.1.1 guide_usb_initialize	4
1.1.2 guide_usb_openStream	4
1.1.3 guide_usb_closeStream	4
1.1.4 guide_usb_exit.....	5
1.1.5 guide_usb_setLogLevel.....	5
1.2 数据类型	5
1.2.1 enum guide_usb_video_mode_e	5
1.2.2 enum guide_usb_device_status_e	7
1.2.3 struct guide_usb_device_info_t	7
1.2.4 struct guide_usb_frame_data_t	8
1.2.5 enum guide_usb_log_level_e	8
1.2.6 OnDeviceConnectStatusCB	9
1.2.7 OnFrameDataReceivedCB	9
2. GUIDE Lib 开发	10
2.1 开发流程	10
2.1.1 流程图	10
2 参数行协议	11

1. 图像&通信(API)

1.1 API 接口

1.1.1 guide_usb_initialize

```
int guide_usb_initialize (const char* deviceName);
```

功能描述

初始化 USB 设备

输入参数

deviceName: 设备名称, 例如: /dev/video0

返回值

>=0 成功, <0 失败

1.1.2 guide_usb_openStream

```
int guide_usb_openStream(guide_usb_device_info_t* deviceInfo, OnFrameDataReceivedCB  
frameRecvCB, OnDeviceConnectStatusCB connectStatusCB);
```

功能描述

打开视频流

输入参数

deviceInfo: 设备信息, 包括图像宽高、视频模式和设备版本 (有效版本 1、2、3)

frameRecvCB: 视频流回调方法

connectStatusCB: 连接状态回调方法

返回值

>=0 成功, <0 失败

1.1.3 guide_usb_closeStream

```
int guide_usb_closeStream();
```

功能描述

关闭视频流

输入参数

无

返回值

>=0 成功，<0 失败

1.1.4 guide_usb_exit

```
int guide_usb_exit();
```

功能描述

退出，清除数据，释放内存

输入参数

无

返回值

>0 成功，<0 失败

1.1.5 guide_usb_setLogLevel

```
int guide_usb_setLogLevel(int level);
```

功能描述

设置 log 等级和开关

输入参数

level: log 的等级，参见 [guide_usb_log_level_e](#)。

返回值

>0 成功，<0 失败

1.2 数据类型

1.2.1 enum guide_usb_video_mode_e

类型定义

```
typedef enum
```

```
{
```

```
YUV = 0, //YUV
YUV_PARAM = 1, //YUV +参数行
Y16 = 2, //Y16
Y16_PARAM = 3, //Y16+ 参数行
Y16_YUV = 4, //Y16+ YUV
Y16_YUV_PARAM = 5, //Y16+ YUV+参数行
X16 = 6, //不支持
X16_PARAM = 7, //不支持
TMP = 8,
TMP_PARAM = 9,
TMP_YUV = 10,
TMP_PARAM_YUV = 11
} guide_usb_video_mode_e;
```

功能描述

机芯视频模式:**SDK 代码中的视频模式参数必须与机芯视频模式保持一致。**

仅部分机芯支持模式 6~11，含 TMP 数据的视频模式仅 ASIC2.0 机芯支持。

各种模式的应用与区别:

YUV:

一般用于观瞄机芯，图像数据为 YUV422 UYVY 格式。

YUV_PARAM:

可以用于观瞄机芯也可用于测温机芯。如果测温机芯不需要全图温度矩阵，那么该模式最适合，YUV 图像数据用于显示，参数行数据可以获取区域最低、最高、平均以及光标点温度。

Y16:

该模式输出纯 16 位数据，可用于分析。

Y16_PARAM :

该模式输出纯 16 位数据和参数行数据，该模式可用于测温机芯。

Y16_YUV:

输出纯 16 位数据和 YUV 数据，只能用于观瞄机芯，但是用于观瞄机芯带宽需求是 YUV 模式的 2 倍，该模式不建议使用。

Y16_YUV_PARAM:

既可用于观瞄机芯也可以用于测温机芯，在带宽允许的情况

下，若全图温度矩阵是必须，该模式最适合，若全图温度矩阵非必须，则 YUV_PARAM 更适合。

TMP:

该模式输出 16 位温度矩阵数据，解析时每个 16 位数据需/10。

TMP_PARAM:

该模式输出 16 位温度矩阵数据和参数行数据，解析温度矩阵时每个 16 位数据需/10。

TMP_YUV:

输出 16 位温度矩阵数据和 YUV 数据，解析温度矩阵时每个 16 位数据需/10，YUV 图像数据用于显示，该模式不建议使用。

TMP_PARAM_YUV:

用于 ASIC2.0 测温机芯，输出 16 位温度矩阵数据和 YUV 数据，解析温度矩阵时每个 16 位数据需/10，YUV 图像数据用于显示。

1.2.2 enum guide_usb_device_status_e

类型定义

```
typedef enum
{
    DEVICE_CONNECT_OK = 1,           //视频流开
    DEVICE_DISCONNECT_OK = -1,       //视频流关
} guide_usb_device_status_e;
```

功能描述

机芯视频流状态。

1.2.3 struct guide_usb_device_info_t

类型定义

```
typedef struct
{
    int width;                       //图像宽度
    int height;                      //图像高度
    guide_usb_video_mode_e video_mode; //视频模式
    int device_version;              //设备版本，有效值有且仅有 1、2、3
```

```
} guide_usb_device_info_t;
```

功能描述

机芯视频信息，打开设备时需要的信息,根据机芯来配置。

1.2.4 struct guide_usb_frame_data_t

类型定义

typedef struct

```
{  
    int frame_width;                //图像宽度  
    int frame_height;              //图像高度  
    short* frame_src_data;          //y16 数据(ASIC2.0 :温度数据)  
    int frame_src_data_length;      //y16 数据长度(ASIC2.0 :温度数据长度)  
    short* frame_yuv_data;          //yuv422 数据  
    int frame_yuv_data_length;      //yuv422 数据长度  
    short* paramLine;               //参数行  
    int paramLine_length;           //参数行长度  
} guide_usb_frame_data_t;
```

功能描述

图像数据相关数据。

Y16 数据从 frame_src_data 获取，**ASIC2.0 机芯时温度数据 TMP 从 frame_src_data 获取。**

YUV 数据从 frame_yuv_data 获取。

1.2.5 enum guide_usb_log_level_e

类型定义

typedef enum

```
{  
    CLOSE = 0,                //关闭日志  
    LOG_FATALLEER = 1,  
    LOG_ERROR = 3,  
    LOG_WARN = 7,  
}
```

```
LOG_INFO = 15,
```

```
LOG_TEST = 31,
```

```
} guide_usb_log_level_e;
```

功能描述

日志等级设置。

1.2.6 OnDeviceConnectStatusCB

类型定义

```
typedef int ( *OnDeviceConnectStatusCB)(guide_usb_device_status_e deviceStatus);
```

功能描述

视频流连接状态回调方法。

1.2.7 OnFrameDataReceivedCB

类型定义

```
typedef int ( *OnFrameDataReceivedCB)(guide_usb_frame_data_t *pVideoData);
```

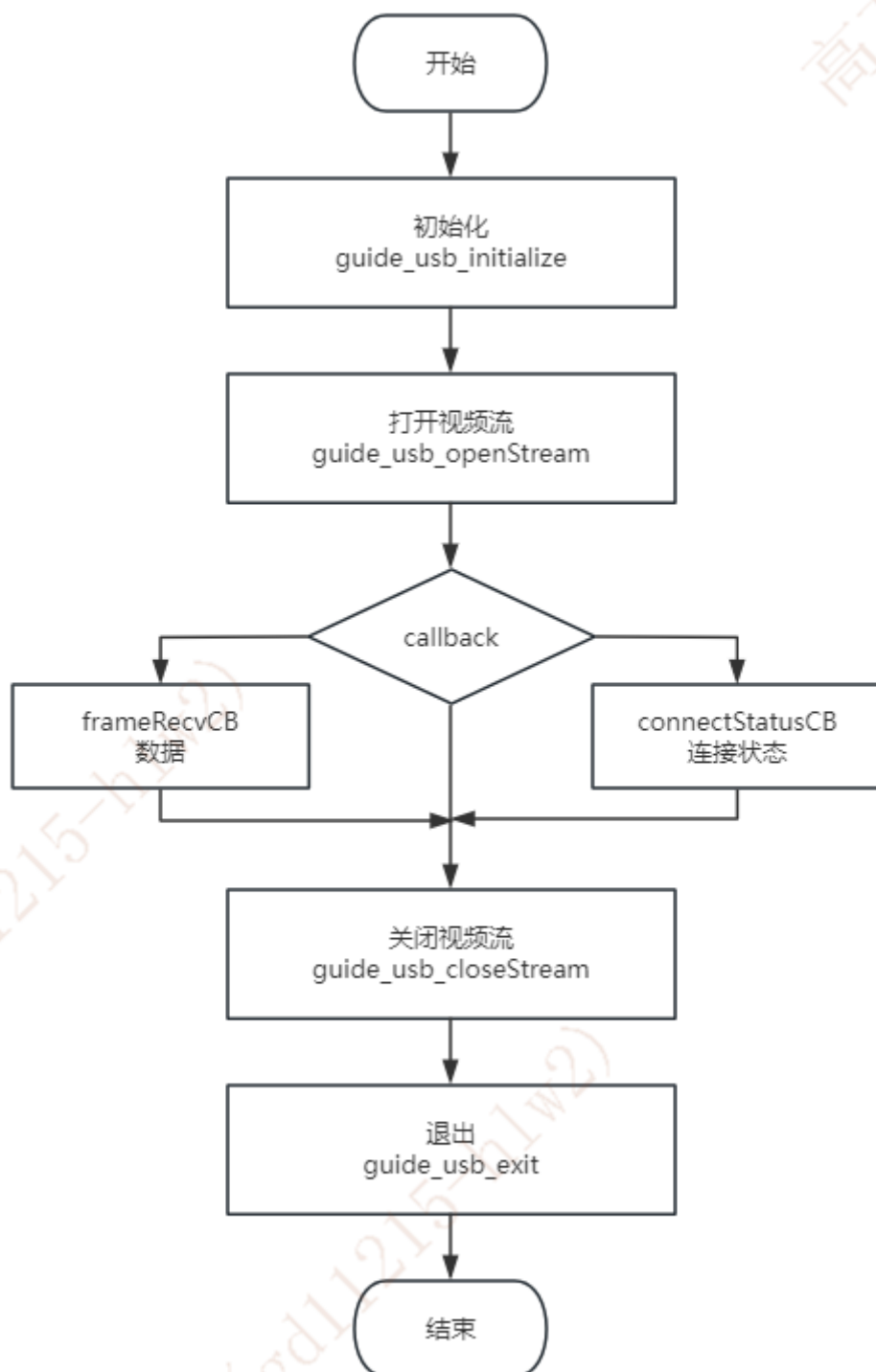
功能描述

视频流回调方法。

2.GUIDE Lib 开发

2.1 开发流程

2.1.1 流程图



2 参数行协议

PARAM 数据是测温接口必须的参数，除了用于测温，该数据还会附带一些信息。

	参数行				
索引	序号	传递内容		所占位宽	说明
0	head1	0x55AA			帧头
1	head2	0x0038			帧头
2	1	测温相关	[15:0]	16	湿度
3	2		[15:0]	16	距离已放大 10 倍
4	3		[15:0]	16	发射率
5	4		[15:0]	16	反射温度
6~27	5~26	预留	[15:0]	16	预留
28	27	快门状态标志	[15:0]	16	0：未打快门，1：正在打快门
29~43	28~42	预留	[15:0]	16	预留
44	43	最热点 X 坐标	[15:0]	16	测出的温度已放大 10 倍
45	44	最热点 Y 坐标	[15:0]	16	
46	45	最热点温度	[15:0]	16	
47	46	最冷点 X 坐标	[15:0]	16	
48	47	最冷点 Y 坐标	[15:0]	16	
49	48	最冷点温度	[15:0]	16	
50	49	光标点 X 坐标	[15:0]	16	
51	50	光标点 Y 坐标	[15:0]	16	
52	51	光标点温度	[15:0]	16	
53	52	区域平均温	[15:0]	16	
54~58	53~57	预留	[15:0]	16	不解析
59	58	0X6666	[15:0]	16	帧尾(不同机芯帧尾位置不同)