



CV186AH Stitch API 使用手册

Version: 1.0.0

Release date: 2023/12

©2022 北京晶视智能科技有限公司
本文件所含信息归北京晶视智能科技有限公司所有。
未经授权，严禁全部或部分复制或披露该等信息。

目录

1	声明	2
2	概述	3
2.1	图片拼接简介	3
2.1.1	图片拼接介绍	3
2.1.2	图片拼接方案示例	4
2.2	功能特点	5
3	功能	7
3.1	Block Overview	7
4	数据结构	8
4.1	数据结构总览	8
4.2	数据结构说明	9
4.2.1	STITCH_ALIGN	9
4.2.2	STITCH_MAX_SRC_NUM	9
4.2.3	STITCH_IP_NUM	9
4.2.4	STITCH_MAX_GRP_NUM	10
4.2.5	STITCH_GRP	10
4.2.6	STITCH_SRC_IDX	11
4.2.7	STITCH_SRC_ATTR_S	11
4.2.8	STITCH_CHN_ATTR_S	12
4.2.9	STITCH_OP_ATTR_S	12
4.2.10	STITCH_WGT_ATTR_S	13
4.2.11	stitch_src_ovlp_attr	14
4.2.12	stitch_src_bd_attr	14
4.2.13	stitch_src_id	15
4.2.14	stitch_wgt_mode	15
4.2.15	stitch_data_src	16
4.2.16	stitch_way_num	17
5	应用程序接口	18
5.1	API 总览	18
5.2	API 流程	19
5.3	API 说明	20
5.3.1	CVI_STITCH_Init	20
5.3.2	CVI_STITCH_DeInit	20
5.3.3	CVI_STITCH_GetAvailableGrp	21
5.3.4	CVI_STITCH_InitGrp	22
5.3.5	CVI_STITCH_DeInitGrp	22
5.3.6	CVI_STITCH_SetSrcAttr	23

5.3.7	CVI_STITCH_GetSrcAttr	24
5.3.8	CVI_STITCH_SetChnAttr	25
5.3.9	CVI_STITCH_GetChnAttr	26
5.3.10	CVI_STITCH_SetOpAttr	26
5.3.11	CVI_STITCH_GetOpAttr	27
5.3.12	CVI_STITCH_SetWgtAttr	28
5.3.13	CVI_STITCH_GetWgtAttr	29
5.3.14	CVI_STITCH_SetRegX	30
5.3.15	CVI_STITCH_EnableGrp	30
5.3.16	CVI_STITCH_DisableGrp	31
5.3.17	CVI_STITCH_Reset	32
5.3.18	CVI_STITCH_SendFrame	33
5.3.19	CVI_STITCH_SendChnFrame	33
5.3.20	CVI_STITCH_GetChnFrame	34
5.3.21	CVI_STITCH_ReleaseChnFrame	35
5.3.22	CVI_STITCH_AttachVbPool	36
5.3.23	CVI_STITCH_DetachVbPool	36
5.3.24	CVI_STITCH_DumpRegInfo	37
5.3.25	CVI_STITCH_Suspend	38
5.3.26	CVI_STITCH_Resume	38
5.3.27	CVI_STITCH_GetDevFd	39
6	错误代码	41
6.1	错误代码	41

修订记录

Revision	Date	Description
1.0.0	2023/08/30	初稿

1 声明



法律声明

本数据手册包含北京晶视智能科技有限公司（下称“晶视智能”）的保密信息。未经授权，禁止使用或披露本数据手册中包含的信息。如您未经授权披露全部或部分保密信息，导致晶视智能遭受任何损失或损害，您应对因之产生的损失/损害承担责任。

本文件内信息如有更改，恕不另行通知。晶视智能不对使用或依赖本文件所含信息承担任何责任。本数据手册和本文件所含的所有信息均按“原样”提供，无任何明示、暗示、法定或其他形式的保证。晶视智能特别声明未做任何适销性、非侵权性和特定用途适用性的默示保证，亦对本数据手册所使用、包含或提供的任何第三方的软件不提供任何保证；用户同意仅向该第三方寻求与此相关的任何保证索赔。此外，晶视智能亦不对任何其根据用户规格或符合特定标准或公开讨论而制作的可交付成果承担责任。

联系我们

地址 北京市海淀区丰豪东路 9 号院中关村集成电路设计园（ICPARK）1 号楼

深圳市宝安区福海街道展城社区会展湾云岸广场 T10 栋

电话 +86-10-57590723 +86-10-57590724

邮编 100094（北京）518100（深圳）

官方网站 <https://www.sophgo.com/>

技术论坛 <https://developer.sophgo.com/forum/index.html>

2 概述

2.1 图片拼接简介

2.1.1 图片拼接介绍

图片拼接 (Image Stitching) 硬件模块是用于加速影像拼接的相关计算：对多路图像进行全景拼接，并且按照指定的投影模式输出图像。

整个拼接过程主要包括标定以及拼接输出。

当多路图像输入 Image Stitching 模块前，或需要通过其他模块对多路图片进行影像几何畸变校正 (GDC/DWA), 抠图 (Crop), 图像缩放 (resize) 等操作。

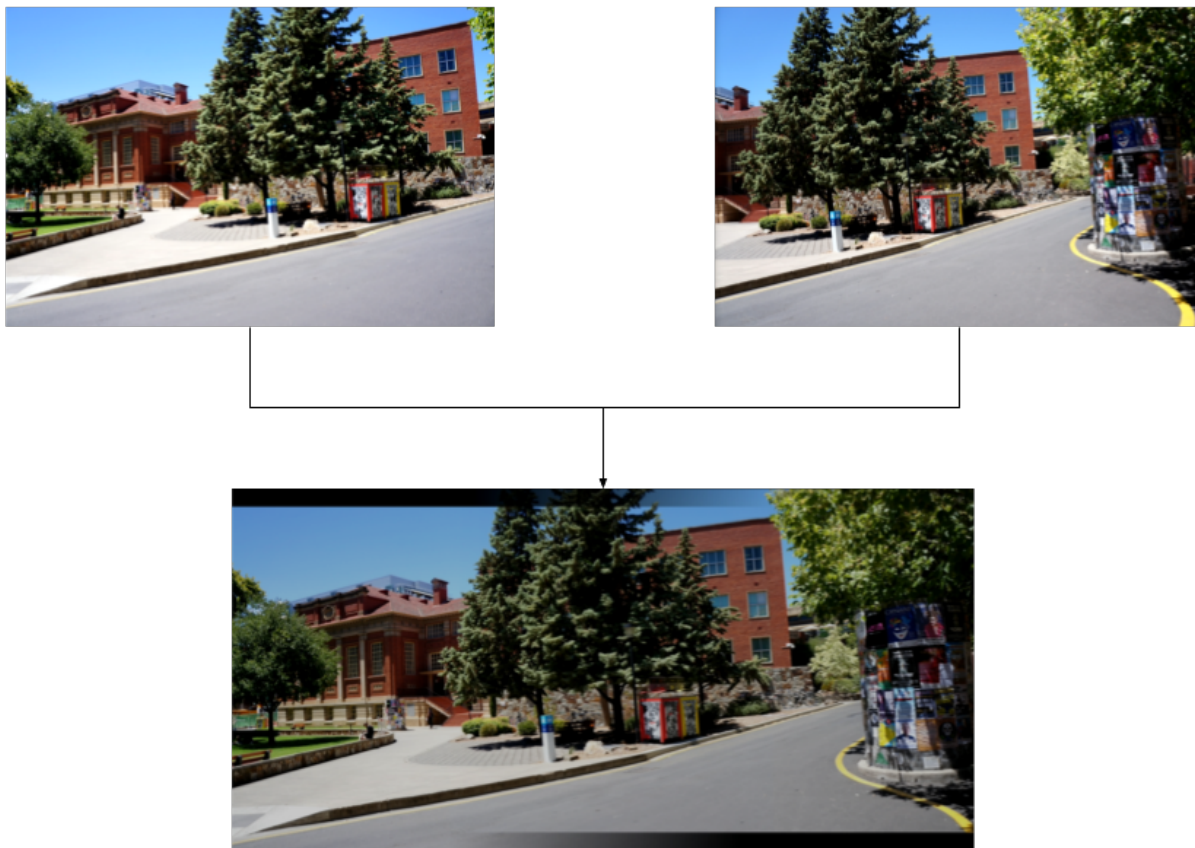


图 2.1: Image Stitching 模块处理双路拼接示例

图 2.1 中，Image Stitching 模块接受双路图片作为输入，通过 Image Stitching 算法根据相同的图像区间，将这些图片拼接在一起。

2.1.2 图片拼接方案示例

对于有着不同图片输入数量的图片拼接任务，拼接方式有多种方案。

图 2.2 以及 图 2.3 分别展示了常用的 2 路与 4 路图片拼接方案。

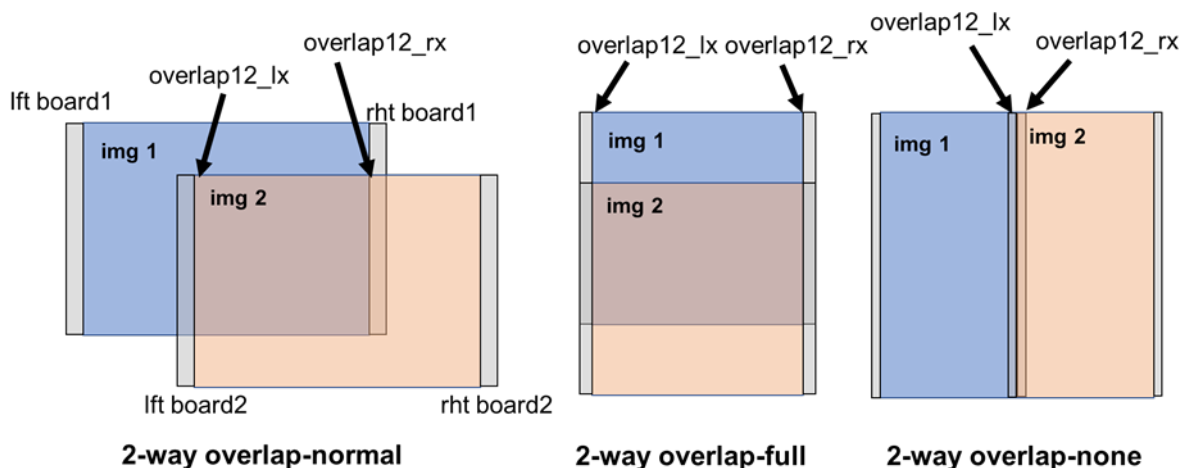


图 2.2: Image Stitching 模块双路图片拼接方案示例

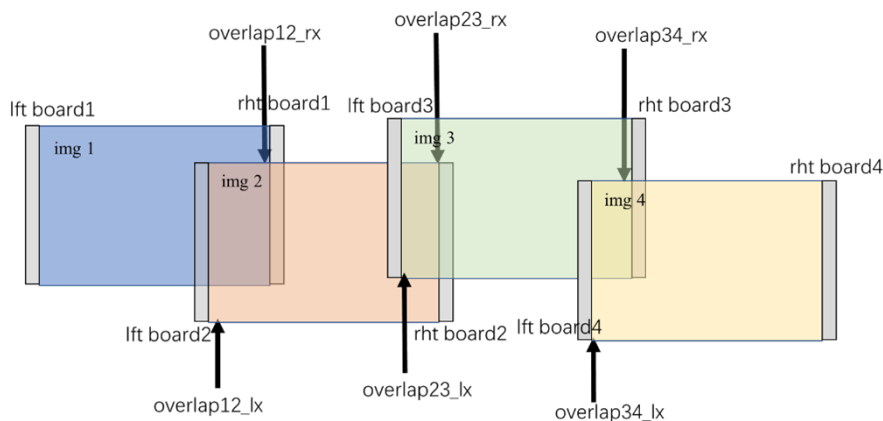


图 2.3: Image Stitching 模块四路图片拼接方案示例

2.2 功能特点

- 配置从端口: 32-bit APB4
- 数据访问主端口: 128-bit AXI4
- 最大分辨率: 4608x2160
- 最小分辨率: 64x64
- 输入源格式来自于 VPSS online 方式, 暂且不支持)
 - RGB888 planar
 - YUV444 planar
- 输入源格式来自于 DRAM:
 - YUV420 planar

- YUV422 planar
- RGB888 planar
- YUV444 planar
- Y only

- **输出格式:**

- YUV420 planar
- YUV422 planar
- RGB888 planar
- YUV444 planar
- Y only

- 支持 alpha-beta 混合 (8 bit 位深)

- **权重格式:(8 bit 位深)**

- YUV 共享 alpha 和 beta (1 alpha + 1 beta)
- UV 共享 alpha 和 beta (2 alpha + 2 beta)

注意: 目前 STITCH 模块暂未支持 VPSS ONLINE

且 STITCH 模块不支持 Color Space Conversion, 也就是说 input 像素格式和 output 像素格式是一致的

3 功能

3.1 Block Overview

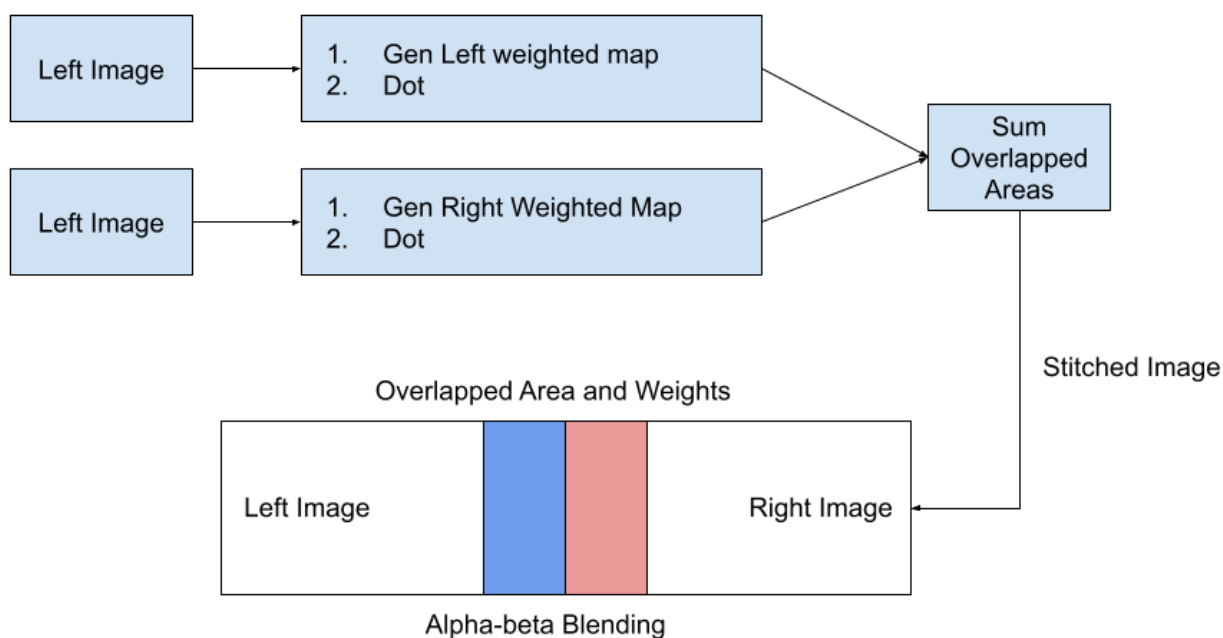


图 3.1: Image Stitching 算法示例

图 3.1 展示了基于 Image Stitching 模块的 2 路图片拼接的原理。

在进行拼接之前，用户需要指定重叠区域及左右权重图（weighted map）；

接着，将左右权重图分别与输入的左右图像进行相乘；

最后，将结果相加以获得最终的图片拼接结果。

4 数据结构

4.1 数据结构总览

· Image Stitching 模块主要提供了以下数据结构:

- 宏定义:
 - * `STITCH_ALIGN` : stitch 的图像尺寸对齐要求
 - * `STITCH_MAX_SRC_NUM` : stitch 的最大输入源数量
 - * `STITCH_IP_NUM` : stitch 的 IP 数量
 - * `STITCH_MAX_GRP_NUM` : stitch 的最大组数量
- 整型 (CVI_S32) :
 - * `STITCH_GRP` : 定义组 ID
 - * `STITCH_SRC_IDX` : 定义源 ID
- 结构体 (Struct):
 - * `STITCH_SRC_ATTR_S` : 定义源属性
 - * `STITCH_CHN_ATTR_S` : 定义 Channel 属性
 - * `STITCH_OP_ATTR_S` : 定义操作属性
 - * `STITCH_WGT_ATTR_S` : 定义权重属性
 - * `struct stitch_src_ovlp_attr` : 定义重叠区域属性
 - * `stitch_src_bd_attr` : 定义图片的左右黑边宽度
- 枚举 (enum):
 - * `stitch_src_id` : 定义源 ID
 - * `stitch_wgt_mode` : 定义权重模式
 - * `stitch_data_src` : 定义数据源
 - * `stitch_way_num` : 定义参与拼接的源数量

4.2 数据结构说明

4.2.1 STITCH_ALIGN

【描述】

定义 stitch 的图像尺寸对齐要求。

【定义】

```
#define STITCH_ALIGN (16)
```

【注意事项】

无

【相关类型及接口】

无

4.2.2 STITCH_MAX_SRC_NUM

【描述】

定义 stitch 的最大输入源数量。

【定义】

```
#define STITCH_MAX_SRC_NUM (4)
```

【注意事项】

无

【相关类型及接口】

无

4.2.3 STITCH_IP_NUM

【描述】

定义 stitch 的 IP 数量。

【定义】

```
#define STITCH_IP_NUM (1)
```

【注意事项】

无

【相关类型及接口】

无

4.2.4 STITCH_MAX_GRP_NUM

【描述】

定义 stitch 的最大组数量。

【定义】

```
#define STITCH_MAX_GRP_NUM (16)
```

【注意事项】

无

【相关类型及接口】

无

4.2.5 STITCH_GRP

【描述】

定义 Stitch GRP ID。STITCH_GRP 表示组 ID，每组可以对应 2way 或者 4way 输入信息，每组只能对应 1 路输出。

【定义】

```
typedef CVI_S32 STITCH_GRP;
```

【成员】

无

【注意事项】

无

【相关类型及接口】

- CVI_STITCH_InitGrp
- CVI_STITCH_DeInitGrp
- CVI_STITCH_GetAvailableGrp
- CVI_STITCH_EnableGrp
- CVI_STITCH_DisableGrp

4.2.6 STITCH_SRC_IDX

【描述】

定义 Stitch SRC ID。STITCH_SRC_IDX 表示原始图像的 id，0 表示第 0 张图，1 表示第一张图

【定义】

```
typedef CVI_S32 STITCH_SRC_IDX;
```

【成员】

无

【注意事项】

无

【相关类型及接口】

- CVI_STITCH_SendFrame
- stitch_src_id

4.2.7 STITCH_SRC_ATTR_S

【描述】

定义 Stitch 源属性。

【定义】

```
typedef struct stitch_src_attr{  
    struct stitch_src_ovlp_attr ovlp_attr;  
    struct stitch_src_bd_attr bd_attr;  
    SIZE_S size[STITCH_MAX_SRC_NUM];  
    PIXEL_FORMAT_E fmt_in;  
    enum stitch_way_num way_num;  
} STITCH_SRC_ATTR_S;
```

【成员】

表 4.1: STITCH_SRC_ATTR_S 成员表

成员名称	描述
stitch_src_ovlp_attr	输入源重叠区域属性
stitch_src_bd_attr	输入源左右边界区域
size	输入源大小
fmt_in	输入源格式
way_num	输入源数目

【注意事项】

无

【相关类型及接口】

- CVI_STITCH_SetSrcAttr
- CVI_STITCH_GetSrcAttr
- CVI_STITCH_GetSrcAttr

4.2.8 STITCH_CHN_ATTR_S

【描述】

定义 STITCH_CHN_ATTR_S 属性。

【定义】

```
typedef struct stitch_chn_attr {  
    SIZE_S size;  
    PIXEL_FORMAT_E fmt_out;  
} STITCH_CHN_ATTR_S;
```

【成员】

表 4.2: STITCH_CHN_ATTR_S 成员表

成员名称	描述
size	输出源大小
fmt_out	输出源格式

【注意事项】

无

【相关类型及接口】

- CVI_STITCH_SetChnAttr
- CVI_STITCH_GetChnAttr

4.2.9 STITCH_OP_ATTR_S

【描述】

定义操作属性。

【定义】

```
typedef struct stitch_op_attr {  
    enum stitch_wgt_mode wgt_mode;  
    enum stitch_data_src data_src;  
} STITCH_OP_ATTR_S;
```

【成员】

表 4.3: STITCH_OP_ATTR_S 成员表

成员名称	描述
stitch_wgt_mode	权重模式
stitch_data_src	数据源

【注意事项】

无

【相关类型及接口】

- CVI_STITCH_SetOptAttr
- CVI_STITCH_GetOptAttr

4.2.10 STITCH_WGT_ATTR_S

【描述】

定义权重的属性属性。

【定义】

```
typedef struct stitch_bld_wgt_attr {  
    u64 phy_addr_wgt[STITCH_MAX_SRC_NUM - 1][2];  
    SIZE_S size_wgt[STITCH_MAX_SRC_NUM - 1];  
} STITCH_WGT_ATTR_S;
```

【成员】

表 4.4: STITCH_WGT_ATTR_S 成员表

成员名称	描述
phy_addr_wgt	权重地址
size_wgt	权重图大小

【注意事项】

无

【相关类型及接口】

- CVI_STITCH_SetWgtAttr
- CVI_STITCH_GetWgtAttr

4.2.11 stitch_src_ovlp_attr

【描述】

定义重叠区域属性。

【定义】

```
struct stitch_src_ovlp_attr {  
    short ovlp_lx[STITCH_MAX_SRC_NUM - 1];  
    short ovlp_rx[STITCH_MAX_SRC_NUM - 1];  
};
```

【成员】

表 4.5: stitch_src_ovlp_attr 成员表

成员名称	描述
ovlp_lx	重叠区域左边界点 x 坐标
ovlp_rx	重叠区域右边界点 x 坐标

【注意事项】

- 对于左边非重叠，右边非重叠以及重叠区域要求按照宽度 4 byte 对齐。
- 无高度对齐限制。

【相关类型及接口】

- [CVI_STITCH_GetOptAttr](#)

4.2.12 stitch_src_bd_attr

【描述】

定义左右图的黑边宽度。

【定义】

```
struct stitch_src_bd_attr {  
    short bd_lx[STITCH_MAX_SRC_NUM];  
    short bd_rx[STITCH_MAX_SRC_NUM];  
};
```

【成员】

表 4.6: stitch_src_bd_attr 成员表

成员名称	描述
bd_lx	左侧图片的黑边宽度
bd_rx	右侧图片的黑边宽度

【注意事项】

无

【相关类型及接口】

- `CVI_STITCH_GetOptAttr`

4.2.13 `stitch_src_id`

【描述】

定义来源的 ID。

【定义】

```
enum stitch_src_id {  
    STITCH_SRC_ID_0 = 0,  
    STITCH_SRC_ID_1,  
    STITCH_SRC_ID_2,  
    STITCH_SRC_ID_3,  
    STITCH_SRC_ID_SEP,  
};
```

【成员】

表 4.7: `stitch_src_id` 成员表

成员名称	描述
<code>STITCH_SRC_ID_0</code>	来源 ID 为 0
<code>STITCH_SRC_ID_1</code>	来源 ID 为 1
<code>STITCH_SRC_ID_2</code>	来源 ID 为 2
<code>STITCH_SRC_ID_3</code>	来源 ID 为 3
<code>STITCH_SRC_ID_SEP</code>	来源 ID 为边界值

【注意事项】

无

【相关类型及接口】

- `STITCH_SRC_IDX`

4.2.14 `stitch_wgt_mode`

【描述】

定义权重图的模式。

【定义】

```
enum stitch_wgt_mode {  
    STITCH_WGT_YUV_SHARE = 0,  
    STITCH_WGT_UV_SHARE,  
    STITCH_WGT_SEP,  
};
```

【成员】

表 4.8: stitch_wgt_mode 成员表

成员名称	描述
STITCH_WGT_YUV_SHARE	权重模式为 YUV 共享, 表示 yuv 3 chn 共享同一字节权重数据
STITCH_WGT_UV_SHARE	权重模式为 UV 共享, 表示 y chn 独占 1 字节数据, yv 共享一字节数据
STITCH_WGT_SEP	权重模式为 SEP

【注意事项】

无

【相关类型及接口】

· STITCH_OP_ATTR_S

4.2.15 stitch_data_src

【描述】

定义数据的来源。

【定义】

```
enum stitch_data_src {  
    STITCH_DATA_SRC_DDR = 0,  
    STITCH_DATA_SRC_VPSS,  
    STITCH_DATA_SRC_SEP,  
};
```

【成员】

表 4.9: stitch_data_src 成员表

成员名称	描述
STITCH_DATA_SRC_DDR	来源为 DDR
STITCH_DATA_SRC_VPSS	来源为 VPSS
STITCH_DATA_SRC_SEP	来源为 SEP

【注意事项】

无

【相关类型及接口】

· STITCH_OP_ATTR_S

4.2.16 stitch_way_num

【描述】

定义 stitch 来源的数量。

【定义】

```
enum stitch_way_num {  
    STITCH_2_WAY = 0,  
    STITCH_4_WAY,  
    STITCH_WAY_SEP,  
};
```

【成员】

表 4.10: stitch_way_num 成员表

成员名称	描述
STITCH_2_WAY	路拼接数量
STITCH_4_WAY	四路拼接数量
STITCH_WAY_SEP	四路拼接数量为 SEP

【注意事项】

无

【相关类型及接口】

- STITCH_SRC_ATTR_S

5 应用程序接口

5.1 API 总览

- Image Stitching 模块主要为用户提供以下 API:
 - CVI_STITCH_Init : 初始化 stitch 模块
 - CVI_STITCH_DeInit : 反初始化 stitch 模块
 - CVI_STITCH_GetAvailableGrp : 获取可用的 stitch 组 ID
 - CVI_STITCH_InitGrp : 初始化 stitch 组 ID
 - CVI_STITCH_DeInitGrp : 反初始化 stitch 组 ID
 - CVI_STITCH_SetSrcAttr : 设置 stitch 模块源属性
 - CVI_STITCH_GetSrcAttr : 获取 stitch 模块源属性
 - CVI_STITCH_SetChnAttr : 设置 stitch 模块 channel 属性
 - CVI_STITCH_GetChnAttr : 获得 stitch 模块 channel 属性
 - CVI_STITCH_SetOpAttr : 设置操作属性
 - CVI_STITCH_GetOpAttr : 获得操作属性
 - CVI_STITCH_SetWgtAttr : 设置权重属性
 - CVI_STITCH_GetWgtAttr : 获得权重属性
 - CVI_STITCH_SetRegX : 设置权重位宽
 - CVI_STITCH_EnableGrp : 启用 stitch 组 ID
 - CVI_STITCH_DisableGrp : 禁用 stitch 组 ID
 - CVI_STITCH_Reset : 重置 stitch 设备
 - CVI_STITCH_SendFrame : 发送源 frame
 - CVI_STITCH_SendChnFrame : 发送输出通道 frame
 - CVI_STITCH_GetChnFrame : 从 channel 获得 frame
 - CVI_STITCH_ReleaseChnFrame : 释放 channel 的 frame
 - CVI_STITCH_AttachVbPool : 使用指定的 vbpool

- CVI_STITCH_DetachVbPool : 取消选定 vbpool
- CVI_STITCH_DumpRegInfo : 输出 reginfo
- CVI_STITCH_Suspend : 挂起设备
- CVI_STITCH_Resume : 恢复设备
- CVI_STITCH_GetDevFd : 获得设备句柄

5.2 API 流程

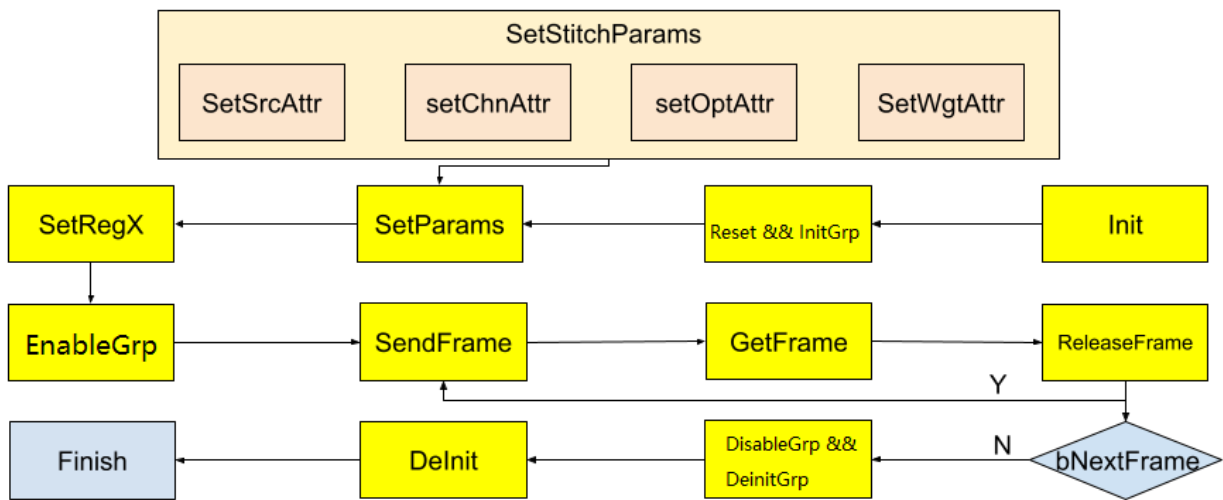


图 5.1: 应用层函数调用流程

在使用 Stitch 处理进行多路拼接时，需要遵循以下流程：首先，通过 CVI_STITCH_Init 和 CVI_STITCH_Reset 对设备进行初始化与重置，CVI_Stitch_InitGrp 创建组 ID；

并通过 CVI_STITCH_SetSrcAttr, CVI_STITCH_SetChnAttr, CVI_STITCH_SetOptAttr, CVI_STITCH_SetWgtAttr 对拼接源，Channel，操作以及权重属性进行设置；

接着，通过 CVI_Stitch_EnableGrp 创建组 ID，CVI_Stitch_SendFrame 发送输入的视频帧数据；

通过 CVI_Stitch_GetFrame 获取处理后的视频帧数据；通过 CVI_Stitch_ReleaseFrame 释放缓冲区的视频帧数据；

如果存在多个输入，重复执行送图取图过程；

最后，通过 CVI_Stitch_DisableGrp 禁用组 ID，CVI_Stitch_DeinitGrp 反初始化组 ID，并通过 CVI_STITCH_Deinit 反初始化 Stitch。

5.3 API 说明

5.3.1 CVI_STITCH_Init

【描述】

初始化 stitch 模块。

【语法】

```
CVI_S32 CVI_STITCH_Init();
```

【参数】

无

【返回值】

表 5.1: CVI_STITCH_Init 返回值表

参数名称	描述
CVI_SUCCESS	初始化成功
其他值	失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- CVI_STITCH_Init 应在进行图片拼接任务最开始调用。

【举例】

```
int s32Ret = CVI_STITCH_Init();
```

5.3.2 CVI_STITCH_DeInit

【描述】

反初始化 stitch 模块。

【语法】

```
CVI_S32 CVI_STITCH_DeInit();
```

【参数】

无

【返回值】

表 5.2: CVI_STITCH_DeInit 返回值表

参数名称	描述
CVI_SUCCESS	反初始化成功
其他值	反初始化失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- CVI_STITCH_DeInit 应在进行图片拼接任务完成后调用。

【举例】

```
s32Ret = CVI_STITCH_DeInit();
```

5.3.3 CVI_STITCH_GetAvailableGrp

【描述】

获取可用组 ID。

【语法】

```
STITCH_GRP CVI_STITCH_GetAvailableGrp(void);
```

【参数】

无

【返回值】

表 5.3: CVI_STITCH_GetAvailableGrp 返回值表

参数名称	描述
STITCH_GRP	组 ID(正常范围是 [0~STITCH_MAX_GRP_NUM-1])

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【举例】

```
s32Ret = CVI_STITCH_GetAvailableGrp();
```


5.3.4 CVI_STITCH_InitGrp

【描述】

初始化 stitch 组 ID。

【语法】

```
CVI_S32 CVI_STITCH_InitGrp(STITCH_GRP stitchGrpIdx);
```

【参数】

表 5.4: CVI_STITCH_SetSrcAttr 参数表

参数名称	描述	输入/输出
stitchGrpIdx	STITCH 组 ID, 该参数范围是 [0~STITCH_MAX_GRP_NUM-1]	输入

【返回值】

表 5.5: CVI_STITCH_InitGrp 返回值表

参数名称	描述
CVI_SUCCESS	初始化组成功
其他值	初始化组失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- CVI_STITCH_InitGrp 不可用无效组 ID。

【举例】

```
s32Ret = CVI_STITCH_InitGrp(grp);
```

5.3.5 CVI_STITCH_DeInitGrp

【描述】

反初始化 stitch 组 ID。

【语法】

```
CVI_S32 CVI_STITCH_DeInitGrp(STITCH_GRP stitchGrpIdx);
```

【参数】

表 5.6: CVI_STITCH_SetSrcAttr 参数表

参数名称	描述	输入/输出
stitchGrpIdx	STITCH 组 ID, 该参数范围是 [0~STITCH_MAX_GRP_NUM-1]	输入

【返回值】

表 5.7: CVI_STITCH_DeInitGrp 返回值表

参数名称	描述
CVI_SUCCESS	初始化组成功
其他值	初始化组失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- CVI_STITCH_DeInitGrp 不可用无效组 ID。

【举例】

```
s32Ret = CVI_STITCH_DeInitGrp(grp);
```

5.3.6 CVI_STITCH_SetSrcAttr

【描述】

设置 stitch 模块源属性。

【语法】

```
CVI_S32 CVI_STITCH_SetSrcAttr(STITCH_GRP stitchGrpIdx, STITCH_SRC_
↪ATTR_S *srcAttr);
```

【参数】

表 5.8: CVI_STITCH_SetSrcAttr 参数表

参数名称	描述	输入/输出
stitchGrpIdx	STITCH 组 ID, 该参数范围是 [0~STITCH_MAX_GRP_NUM-1]	输入
srcAttr	STITCH 源属性, 该参数不能为空	输入

【返回值】

表 5.9: CVI_STITCH_SetSrcAttr 返回值表

参数名称	描述
CVI_SUCCESS	设置 Stitch 源参数成功
其他值	设置 Stitch 源参数失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- 需要在 CVI_STITCH_InitGrp 函数调用后启动。

【举例】

```
CVI_STITCH_SetSrcAttr(0, srcAttr);
```

5.3.7 CVI_STITCH_GetSrcAttr

【描述】

获取 stitch 模块源属性。

【语法】

```
CVI_S32 CVI_STITCH_GetSrcAttr(STITCH_GRP stitchGrpIdx, STITCH_SRC_
↪ATTR_S *srcAttr);
```

【参数】

表 5.10: CVI_STITCH_GetSrcAttr 参数表

参数名称	描述	输入/输出
stitchGrpIdx	STITCH 组 ID, 该参数范围是 [0~STITCH_MAX_GRP_NUM-1]	输入
srcAttr	STITCH 源属性, 该参数不能为空	输出

【返回值】

表 5.11: CVI_STITCH_GetSrcAttr 返回值表

参数名称	描述
CVI_SUCCESS	获得 Stitch 源参数成功
其他值	获得 Stitch 源参数失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- 需要在 CVI_STITCH_Init 函数调用后启动。

【举例】

```
CVI_STITCH_GetSrcAttr(0, srcAttr);
```

5.3.8 CVI_STITCH_SetChnAttr

【描述】

设置 stitch 模块 channel 属性。

【语法】

```
CVI_S32 CVI_STITCH_SetChnAttr(STITCH_GRP stitchGrpIdx, STITCH_CHN_  
↪ATTR_S *chnAttr);
```

【参数】

表 5.12: CVI_STITCH_SetChnAttr 参数表

参数名称	描述	输入/输出
stitchGrpIdx	STITCH 组 ID, 该参数范围是 [0~STITCH_MAX_GRP_NUM-1]	输入
chnAttr	通道属性, 该参数不能为空	输入

【返回值】

表 5.13: CVI_STITCH_SetChnAttr 返回值表

参数名称	描述
CVI_SUCCESS	设置 Channel 属性成功
其他值	设置 Channel 属性失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- 需要在 CVI_STITCH_SetSrcAttr 函数调用后启动。

【举例】

```
CVI_STITCH_SetChnAttr(grp, chnAttr);
```

5.3.9 CVI_STITCH_GetChnAttr

【描述】

获得 stitch 模块 channel 属性。

【语法】

```
CVI_S32 CVI_STITCH_GetChnAttr(STITCH_GRP stitchGrpIdx, STITCH_CHN_
↪ATTR_S *chnAttr);
```

【参数】

表 5.14: CVI_STITCH_GetChnAttr 参数表

参数名称	描述	输入/输出
stitchGrpIdx	STITCH 组 ID, 该参数范围是 [0~STITCH_MAX_GRP_NUM-1]	输入
chnAttr	通道属性, 该参数不能为空	输出

【返回值】

表 5.15: CVI_STITCH_GetChnAttr 返回值表

参数名称	描述
CVI_SUCCESS	获得 Channel 属性成功
其他值	获得 Channel 属性失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- 需要在 CVI_STITCH_Init 函数调用后启动。

【举例】

```
CVI_STITCH_GetChnAttr(grp, chnAttr);
```

5.3.10 CVI_STITCH_SetOpAttr

【描述】

设置操作属性。

【语法】

```
CVI_S32 CVI_STITCH_SetOpAttr(STITCH_GRP stitchGrpIdx, STITCH_OP_ATTR_
↪S *opAttr);
```

【参数】

表 5.16: CVI_STITCH_SetOpAttr 参数表

参数名称	描述	输入/输出
stitchGrpIdx	STITCH 组 ID, 该参数范围是 [0~STITCH_MAX_GRP_NUM-1]	输入
opAttr	操作属性, 该参数不能为空	输入

【返回值】

表 5.17: CVI_STITCH_SetOpAttr 返回值表

参数名称	描述
CVI_SUCCESS	设置操作属性成功
其他值	设置操作属性失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- 需要在 CVI_STITCH_Init 函数调用后启动。

【举例】

```
CVI_STITCH_SetOpAttr(grp, opAttr);
```

5.3.11 CVI_STITCH_GetOpAttr

【描述】

获得操作属性。

【语法】

```
CVI_S32 CVI_STITCH_GetOpAttr(STITCH_GRP stitchGrpIdx, STITCH_OP_
↪ATTR_S *opAttr);
```

【参数】

表 5.18: CVI_STITCH_GetOpAttr 参数表

参数名称	描述	输入/输出
stitchGrpIdx	STITCH 组 ID, 该参数范围是 [0~STITCH_MAX_GRP_NUM-1]	输入
opAttr	操作属性, 该参数不能为空	输出

【返回值】

表 5.19: CVI_STITCH_GetOpAttr 返回值表

参数名称	描述
CVI_SUCCESS	获得操作属性成功
其他值	获得操作属性失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- 需要在 CVI_STITCH_Init 函数调用后启动。

【举例】

```
CVI_STITCH_GetOpAttr(grp, opAttr)
```

5.3.12 CVI_STITCH_SetWgtAttr

【描述】

设置权重属性。

【语法】

```
CVI_S32 CVI_STITCH_SetWgtAttr(STITCH_GRP stitchGrpIdx, STITCH_WGT_
↪ATTR_S *wgtAttr);
```

【参数】

表 5.20: CVI_STITCH_SetWgtAttr 参数表

参数名称	描述	输入/输出
stitchGrpIdx	STITCH 组 ID, 该参数范围是 [0~STITCH_MAX_GRP_NUM-1]	输入
wgtAttr	权重属性, 该参数不能为空	输入

【返回值】

表 5.21: CVI_STITCH_SetWgtAttr 返回值表

参数名称	描述
CVI_SUCCESS	设置权重属性成功
其他值	设置权重属性失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- 需要在 CVI_STITCH_SetChnAttr 函数调用后启动，权重属性参数依赖源属性和输出通道属性。

【举例】

```
CVI_STITCH_SetWgtAttr(grp, wgtAttr);
```

5.3.13 CVI_STITCH_GetWgtAttr

【描述】

获得权重属性。

【语法】

```
CVI_S32 CVI_STITCH_GetWgtAttr(STITCH_GRP stitchGrpIdx, STITCH_WGT_
↪ATTR_S *wgtAttr);
```

【参数】

表 5.22: CVI_STITCH_GetWgtAttr 参数表

参数名称	描述	输入/输出
stitchGrpIdx	STITCH 组 ID，该参数范围是 [0~STITCH_MAX_GRP_NUM-1]	输入
wgtAttr	权重属性，该参数不能为空	输出

【返回值】

表 5.23: CVI_STITCH_GetWgtAttr 返回值表

参数名称	描述
CVI_SUCCESS	获得权重属性成功
其他值	获得重属性败，返回错误码。错误码定义参考表 1

【需求】

- 头文件：cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- 需要在 CVI_STITCH_Init 函数调用后启动。

【举例】

```
CVI_STITCH_GetWgtAttr(grp, wgtAttr);
```


5.3.14 CVI_STITCH_SetRegX

【描述】

设置权重位宽。

【语法】

```
CVI_S32 CVI_STITCH_SetRegX(CVI_U8 regX);
```

【参数】

表 5.24: CVI_STITCH_SetRegX 参数表

参数名称	描述	输入/输出
regX	权重位宽，默认为 16	输入

【返回值】

表 5.25: CVI_STITCH_SetRegX 返回值表

参数名称	描述
CVI_SUCCESS	设置权重位宽成功
其他值	设置权重位宽失败，返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- 需要在 CVI_STITCH_Init 函数调用后启动。

【举例】

```
s32Ret = CVI_STITCH_SetRegX(16);
```

5.3.15 CVI_STITCH_EnableGrp

【描述】

启用 stitch 设备。

【语法】

```
CVI_S32 CVI_STITCH_EnableGrp(STITCH_GRP stitchGrpIdx);
```

【参数】

表 5.26: CVI_STITCH_EnableGrp 参数表

参数名称	描述	输入/输出
stitchGrpIdx	STITCH 组 ID, 该参数范围是 [0~STITCH_MAX_GRP_NUM-1]	输入

【返回值】

表 5.27: CVI_STITCH_EnableGrp 返回值表

参数名称	描述
CVI_SUCCESS	启用设备成功
其他值	启用设备失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- 需要完成 Stitch 初始化组 ID, 参数设置以及权重位宽设置后启动。

【举例】

```
CVI_STITCH_EnableGrp(0);
```

5.3.16 CVI_STITCH_DisableGrp

【描述】

禁用 stitch 设备。

【语法】

```
CVI_S32 CVI_STITCH_DisableGrp(STITCH_GRP stitchGrpIdx);
```

【参数】

表 5.28: CVI_STITCH_DisableGrp 参数表

参数名称	描述	输入/输出
stitchGrpIdx	STITCH 组 ID, 该参数范围是 [0~STITCH_MAX_GRP_NUM-1]	输入

【返回值】

表 5.29: CVI_STITCH_DisableGrp 返回值表

参数名称	描述
CVI_SUCCESS	禁用设备成功
其他值	禁用设备失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvl_stitch.h, cvl_debug.h, cvl_base.h

【注意】

- 完成图片拼接任务后, 需要在 Stitch_DeInit 之前调用。

【举例】

```
CVI_STITCH_DisableGrp(grp);
```

5.3.17 CVI_STITCH_Reset

【描述】

重置 stitch 设备。

【语法】

```
CVI_S32 CVI_STITCH_Reset(void);
```

【参数】

无

【返回值】

表 5.30: CVI_STITCH_Reset 返回值表

参数名称	描述
CVI_SUCCESS	重置设备成功
其他值	重置设备失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvl_stitch.h, cvl_debug.h, cvl_base.h

【注意】

- 当存在组 ID 使能工作时, 禁止使用 CVI_STITCH_Reset 调用。

【举例】

```
CVI_STITCH_Reset();
```

5.3.18 CVI_STITCH_SendFrame

【描述】

发送源 frame。

【语法】

```
CVI_S32 CVI_STITCH_SendFrame(STITCH_GRP stitchGrpIdx, STITCH_SRC_IDX_  
↪srcIdx, const VIDEO_FRAME_INFO_S *VideoFrame, CVI_S32 MilliSec);
```

【参数】

表 5.31: CVI_STITCH_SendFrame 参数表

参数名称	描述	输入/输出
stitchGrpIdx	STITCH 组 ID, 该参数范围是 [0~STITCH_MAX_GRP_NUM-1]	输入
srcIdx	源 Id 号	输入
VideoFrame	视频数据, 该参数不能为空	输入
MilliSec	超时时间 (毫秒), 该参数不能为空	输入

【返回值】

表 5.32: CVI_STITCH_SendFrame 返回值表

参数名称	描述
CVI_SUCCESS	发送视频帧数据成功
其他值	发送视频帧数据失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- VideoFrame 的图像地址必须根据 VB 的要求进行 16 字节对齐。

【举例】

```
CVI_STITCH_SendFrame(0, src_id, stVideoFrame, timeout_ms);
```

5.3.19 CVI_STITCH_SendChnFrame

【描述】

表示用户配置好 VIDEO_FRAME_INFO_S 输出图像, 发送到指定的组 ID。

【语法】

```
CVI_S32 CVI_STITCH_SendChnFrame(STITCH_GRP stitchGrpIdx, const VIDEO_  
↪FRAME_INFO_S *VideoFrame, CVI_S32 MilliSec);
```

【参数】

表 5.33: CVI_STITCH_SendChnFrame 参数表

参数名称	描述	输入/输出
stitchGrpIdx	STITCH 组 ID, 该参数范围是 [0~STITCH_MAX_GRP_NUM-1]	输入
VideoFrame	视频帧数据, 该参数不能为空	输入
MilliSec	超时时间 (毫秒), 该参数不能为空	输入

【返回值】

表 5.34: CVI_STITCH_SendChnFrame 返回值表

参数名称	描述
CVI_SUCCESS	发送视频帧数据成功
其他值	发送视频帧数据失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- 使用此 api 时, sdk 内部不会自动为输出图像分配内存, VideoFrame 来源于用户。

【举例】

```
CVI_STITCH_SendChnFrame(grp, stVideoFrameOut, timeout_ms);
```

5.3.20 CVI_STITCH_GetChnFrame

【描述】

从 channel 获得视频帧。

【语法】

```
CVI_S32 CVI_STITCH_GetChnFrame(STITCH_GRP stitchGrpIdx, VIDEO_FRAME_
↪INFO_S *VideoFrame, CVI_S32 MilliSec);
```

【参数】

表 5.35: CVI_STITCH_GetChnFrame 参数表

参数名称	描述	输入/输出
stitchGrpIdx	STITCH 组 ID, 该参数范围是 [0~STITCH_MAX_GRP_NUM-1]	输入
VideoFrame	视频帧数据, 该参数不能为空	输出
MilliSec	超时时间 (毫秒), 该参数不能为空	输入

【返回值】

表 5.36: CVI_STITCH_GetChnFrame 返回值表

参数名称	描述
CVI_SUCCESS	从 channel 获得视频帧数据成功
其他值	从 channel 获得视频帧数据失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【举例】

```
CVI_STITCH_GetChnFrame(grp, stVideoFrameOut, timeout_ms);
```

5.3.21 CVI_STITCH_ReleaseChnFrame

【描述】

释放 channel 的 frame。

【语法】

```
CVI_S32 CVI_STITCH_ReleaseChnFrame(STITCH_GRP stitchGrpIdx, VIDEO_
↪FRAME_INFO_S *VideoFrame);
```

【参数】

表 5.37: CVI_STITCH_ReleaseChnFrame 参数表

参数名称	描述	输入/输出
stitchGrpIdx	STITCH 组 ID, 该参数范围是 [0~STITCH_MAX_GRP_NUM-1]	输入
VideoFrame	视频帧数据, 该参数不能为空	输入

【返回值】

表 5.38: CVI_STITCH_ReleaseChnFrame 返回值表

参数名称	描述
CVI_SUCCESS	从 Channel 释放缓冲区的视频帧数据成功
其他值	从 Channel 释放缓冲区的视频帧数据失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【举例】

```
CVI_STITCH_ReleaseChnFrame(grp, stVideoFrameOut);
```

5.3.22 CVI_STITCH_AttachVbPool

【描述】

使用指定的 vbpool。

【语法】

```
CVI_S32 CVI_STITCH_AttachVbPool(STITCH_GRP stitchGrpIdx, VB_POOL_
↪ VbPool);
```

【参数】

表 5.39: CVI_STITCH_AttachVbPool 参数表

参数名称	描述	输入/输出
stitchGrpIdx	STITCH 组 ID, 该参数范围是 [0~STITCH_MAX_GRP_NUM-1]	输入
VbPool	VB Pool 号, 取值范围: [0, VB_MAX_COMM_POOLS]	输入

【返回值】

表 5.40: CVI_STITCH_AttachVbPool 返回值表

参数名称	描述
CVI_SUCCESS	使用指定的 vbpool 成功
其他值	使用指定的 vbpool 失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- 可选操作。
- 在 CVI_STITCH_InitGrp 之后调用。

【举例】

```
CVI_STITCH_AttachVbPool(grp, vbpool);
```

5.3.23 CVI_STITCH_DetachVbPool

【描述】

取消选定 vbpool。

【语法】

```
CVI_S32 CVI_STITCH_DetachVbPool(STITCH_GRP stitchGrpIdx);
```

【参数】

表 5.41: CVI_STITCH_DetachVbPool 参数表

参数名称	描述	输入/输出
stitchGrpIdx	STITCH 组 ID, 该参数范围是 [0~STITCH_MAX_GRP_NUM-1]	输入

【返回值】

表 5.42: CVI_STITCH_DetachVbPool 返回值表

参数名称	描述
CVI_SUCCESS	取消选定 vbpool 成功
其他值	取消选定 vbpool 失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- 使用该 API 前, 需要调用 AttachVbPool 函数。

【举例】

```
CVI_STITCH_DetachVbPool(grp);
```

5.3.24 CVI_STITCH_DumpRegInfo

【描述】

输出 reginfo。

【语法】

```
CVI_S32 CVI_STITCH_DumpRegInfo(void);
```

【参数】

无

【返回值】

表 5.43: CVI_STITCH_DumpRegInfo 返回值表

参数名称	描述
CVI_SUCCESS	输出 reginfo 成功
其他值	输出 reginfo 失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: `cvi_stitch.h`, `cvi_debug.h`, `cvi_base.h`

【注意】

- 可选操作。

【举例】

```
CVI_STITCH_DumpRegInfo();
```

5.3.25 CVI_STITCH_Suspend

【描述】

挂起设备。

【语法】

```
CVI_S32 CVI_STITCH_Suspend(void);
```

【参数】

无

【返回值】

表 5.44: CVI_STITCH_Suspend 返回值表

参数名称	描述
CVI_SUCCESS	挂起设备成功
其他值	挂起设备失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: `cvi_stitch.h`, `cvi_debug.h`, `cvi_base.h`

【注意】

- 使用该 API 将挂起 Stitch 设备。

【举例】

```
CVI_STITCH_Suspend();
```

5.3.26 CVI_STITCH_Resume

【描述】

恢复设备。

【语法】

```
CVI_S32 CVI_STITCH_Resume(void);
```

【参数】

无

【返回值】

表 5.45: CVI_STITCH_Resume 返回值表

参数名称	描述
CVI_SUCCESS	恢复设备成功
其他值	恢复设备失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- 使用该 API 将恢复挂起 Stitch 设备。

【举例】

```
CVI_STITCH_Resume();
```

5.3.27 CVI_STITCH_GetDevFd

【描述】

获得设备句柄。

【语法】

```
CVI_S32 CVI_STITCH_GetDevFd();
```

【参数】

无

【返回值】

表 5.46: CVI_STITCH_GetDevFd 返回值表

参数名称	描述
CVI_SUCCESS	获得设备句柄成功
其他值	获得设备句柄失败, 返回错误码。错误码定义参考表 1

【需求】

- 头文件: cvi_stitch.h, cvi_debug.h, cvi_base.h

【注意】

- 使用该 API 将获得 Stitch 设备的句柄。

【举例】

```
CVI_STITCH_GetDevFd();
```

6 错误代码

6.1 错误代码

表 1 列出了 Stitch 模块的 API 错误代码。

表 6.1: 表 1 Stitch 模块的 API 错误代码表

错误码	宏定义	描述
0xc0228006	CVI_ERR_STITCH_NULL_PTR	使用了空指针
0xc0228010	CVI_ERR_STITCH_NOTREADY	系统尚未准备就绪，可能尚未初始化或加载，打开设备文件失败
0xc0228001	CVI_ERR_STITCH_INVALID_SRCID	不合法的 src ID
0xc0228004	CVI_ERR_STITCH_EXIST	资源存在
0xc0228005	CVI_ERR_STITCH_UNEXIST	资源不存在
0xc0228008	CVI_ERR_STITCH_NOT_SUPPORT	操作或类型在当前情况下不支持
0xc0228009	CVI_ERR_STITCH_NOT_PERM	操作不被允许，例如尝试修改静态属性
0xc022800c	CVI_ERR_STITCH_NOMEM	由于内存分配引起的失败
0xc022800d	CVI_ERR_STITCH_NOBUF	由于缓冲区分配引起的失败
0xc0228003	CVI_ERR_STITCH_ILLEGAL_PARAM	至少存在一个不合法的参数或不合法的枚举值
0xc0228012	CVI_ERR_STITCH_BUSY	资源繁忙
0xc022800e	CVI_ERR_STITCH_BUF_EMPTY	缓冲区为空