

ISP

开发参考

文档版本 V1.7

发布日期 2025-12-12

前言

概述

本文为使用 ISP 开发的程序员而写，目的是为您在开发过程中遇到的问题提供解决办法和帮助。

 说明

未经特殊说明，以 XM7606V13 指代 XM7606 系列所有芯片。以 XM7206V11A 指代 XM7206 系列所有芯片。

产品版本

与本文档相对应的产品版本如下。

产品名称	产品版本
XM7606	V13
XM7206	V11A

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

符号约定

在本文中可能出现下列标志，它们所代表的含义如下。

符号	说明
 危险	表示如不可避免则将会导致死亡或严重伤害的具有高等级风险的危害。
 警告	表示如不可避免则可能导致死亡或严重伤害的具有中等级风险的危害。
 注意	表示如不可避免则可能导致轻微或中度伤害的具有低等级风险的危害。
须知	用于传递设备或环境安全警示信息。如不可避免则可能会导致设备损坏、数据丢失、设备性能降低或其它不可预知的结果。 “须知”不涉及人身伤害。
 说明	对正文中重点信息的补充说明。 “说明”不是安全警示信息，不涉及人身、设备及环境伤害信息。

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。

修订日期	版本	修订说明
2024-12-21	V1.0	DI 版本发布。
2025-03-04	V1.1	1. 更新 CA 结构体，sat_by_y、sat_by_c 曲线改为手动或自动随 ISO 联动可调。 2. 更新 DRC auto 模式结构体，drc 曲线相关参数改为随曝光比联动。 3. 更新 WDR 章节暂未支持参数相关描述。
		1. 更新 aver_root_mode 参数说明。 2. 新增 xmedia_isp_wait_state 接口输入参数 millisec 的单位为 ms。 3. 新增 FPN，DPC 标定通路使用注意事项。 4. 新增 STNR、Sharpen，AFC 模块接口及数据结构说明。

修订日期	版本	修订说明
		5. 缩略语新增 AFC 说明。 6. 格式修订。
		1. AINR 模块新增加载模型 xmedia_isp_load_ainr_model、卸载模型 xmedia_isp_unload_ainr_model 接口说明。 2. proc 调试信息新增算法相关参数及说明。 3. proc 调试信息 AE 部分 int_time_sf, iso_sf 等参数 名称修订。 4. 修改 BLC 模块数据结构名称, 将数据结构中的 black_level 修改为 blc。 5. 新增章节 SENSOR, 描述 SENSOR 相关接口及数 据结构定义。
2025-04-15	V1.2	AE 模块, 更新 xmedia_ae_smart_exposure_attr 注 意事项说明。
		1. 修改 xmedia_isp_unload_ainr_model 接口描述。 2. 修改 xmedia_awb_unregister 接口描述。 3. 修改 xmedia_isp_set_mesh_shading_attr 接口入参 名称及接口定义。 4. 修改 DRC 模块 tm_value_max_user 参数范围为[0, 255]。
		更新 BNR 模块 noise_profile_thd、 noise_profile_coef0、noise_profile_coef1 参数说明。
		删除 DRC 模块 manual 模式中参数 drc_str 的暂未支 持说明, 该参数支持调试。
2025-06-05	V1.3	AINR 模块, 新增模型加载, 卸载以及使能 ainr 功能的 注意事项说明。
		XM7206V11A 不支持 HLC、DPC 模块的静态标定和 校正、Radial Crop、RGB-IR。 XM7206V11A 支持 STNR、Sharpen、AFC。

修订日期	版本	修订说明
2025-08-20	V1.4	1. CSC 模块，将数据结构 xmedia_isp_csc_matrx 更名为 xmedia_isp_csc_matrix。 2. SENSOR 模块，更新 xmedia_sensor_work_mode 枚举定义；删除 xmedia_sensor_set_link_signal 接口描述；新增 xmedia_sensor_create_group、xmedia_sensor_destroy_group 接口及相关数据结构描述。 3. proc 新增 sensor group info 相关字段及描述。
		1. 更新了算法接口的注意事项，描述了不同版本的支持情况。 2. DRC 模块，新增根据版本描述参数取值范围。 3. BNR 模块，数据结构 xmedia_isp_bnr_manual_attr 新增成员 xmedia_u16 snr_wdr_comp_scl;xmedia_u16 tnr_mot_noise_scl_high;数据结构 xmedia_isp_bnr_auto_attr 新增成员 xmedia_u16 snr_wdr_comp_scl[XMEDIA_ISP_ISO_MAX_CO UNT];xmedia_u16 tnr_mot_noise_scl_high[XMEDIA_ISP_ISO_MAX _COUNT];数据结构 xmedia_isp_bnr_attr 新增成 员 xmedia_bool snr_wdr_comp_en; 4. DMS 模块，更新参数描述。 5. FPN 模块，更新参数描述。 6. 新增 LCAC 模块描述。 7. EXPANDER 模块，更新参数描述。 8. STNR 模块，更新参数描述。 9. proc 新增 ANTI_FALSE_COLOR INFO、LCAC INFO、SHARPEN INFO、STNR INFO 相关字段 及描述。
		1. Expander 模块，更新 api 接口使用注意事项说明。 2. Radial Crop 模块，更新参数取值范围及默认值。

修订日期	版本	修订说明
		3. Sensor 模块，更新 xmedia_sensor_property 描述。 4. 将 xmedia_isp_pipe_diff_attr 更名为 xmedia_isp_diff_attr。 5. 新增 xmedia_isp_iris_f_no 数据结构说明。
2025-09-29	V1.5	增加 fpn 标定帧数取值范围的描述。
		1. Proc 信息，新增字段 unused_num、missed_run_num；完善 AE/AWB proc 信息描述。 2. BNR 和 DRC 模块，新增最小分辨率限制说明。 3. 更新接口 xmedia_awb_set_color_sector_attr 使用注意事项。
2025-11-27	V1.6	修改 smart_interval 在注意事项里的描述。
		新增 AINR 的 5 个接口： xmedia_isp_get_ainr_state xmedia_isp_load_model xmedia_isp_unload_model xmedia_isp_bind_model xmedia_isp_unbind_model
		新增 ae 数据结构与 ctrl 函数的描述： xmedia_isp_ae_result xmedia_isp_ae_info pfn_ae_ctrl
		tnr_var_scl 使用含义的更改。
2025-12-12	V1.7	1. 更新接口 xmedia_isp_register_af_lib 定义。 2. 新增数据结构描述： xmedia_isp_af_param xmedia_isp_af_result xmedia_isp_af_info xmedia_isp_af_pdaf_statistics xmedia_isp_fe_af_zone_matrix xmedia_isp_be_af_zone_matrix

修订日期	版本	修订说明
		xmedia_lens_register_info xmedia_lens_register_isp_func xmedia_lens_register_ae_info xmedia_lens_register_ae_func xmedia_lens_register_awb_info xmedia_lens_register_af_info xmedia_lens_register_af_func xmedia_sensor_pdaf_info xmedia_sensor_pdaf_feature xmedia_sensor_pdaf_type xmedia_sensor_pdaf_correction xmedia_sensor_pdaf_pattern xmedia_sensor_pdaf_pattern_type xmedia_sensor_pdaf_point xmedia_isp_point 3. xmedia_sensor_register_isp_func 新增成员： pfn_sensor_get_hmax_vmax pfn_sensor_update_reg_info pfn_sensor_get_pdaf_info pfn_sensor_format_pdaf pfn_sensor_calc_image_size pfn_sensor_low_power 4. 新增 LENS 章节。 5. 新增 AF 章节。 wdr 模型下不允许开启 b3d。

目 录

前 言	i
1 概述	1
1.1 概述	1
1.2 功能描述	1
1.2.1 架构	2
1.2.2 开发模式	2
1.2.3 内部流程	3
1.2.4 软件流程	4
1.2.5 文件组织	6
2 系统控制	8
2.1 功能概述	8
2.2 API 参考	8
2.3 数据类型	37
3 SENSOR	84
3.1 功能概述	84
3.2 API 参考	84
3.3 数据类型	101
4 LENS	125
4.1 功能概述	125
4.2 API 参考	125

5 AE.....	129
5.1 概述	129
5.2 重要概念	130
5.3 功能描述	130
5.4 API 参考	131
5.4.1 AE 库接口	131
5.4.2 AE 控制模块	133
5.5 数据类型	152
5.5.1 Register.....	152
5.5.2 AE	162
6 AWB.....	196
6.1 概述	196
6.2 重要概念	196
6.3 功能描述	196
6.4 API 参考	197
6.4.1 AWB 库接口	197
6.4.2 AWB 控制模块	199
6.5 数据类型	214
6.5.1 Register.....	214
6.5.2 WB	219
7 AF	250
7.1 概述	250
7.2 重要概念	250
7.3 功能描述	250
7.4 API 参考	252
7.4.1 AF 库接口	252
7.4.2 AF 控制模块	254

7.5 数据类型	257
7.5.1 Register.....	257
7.5.2 AF	258
8 IMP	265
8.1 HLC	265
8.1.1 功能描述.....	265
8.1.2 API 参考	265
8.1.3 数据类型.....	267
8.2 Gamma.....	268
8.2.1 功能描述.....	268
8.2.2 API 参考	268
8.2.3 数据类型.....	270
8.3 DRC.....	274
8.3.1 功能描述.....	274
8.3.2 API 参考	275
8.3.3 数据类型.....	277
8.4 Mesh Shading	286
8.4.1 功能描述.....	286
8.4.2 API 参考	287
8.4.3 数据类型.....	291
8.5 DPC.....	294
8.5.1 功能描述.....	294
8.5.2 API 参考	295
8.5.3 数据类型.....	301
8.6 Crosstalk	311
8.6.1 概述.....	311
8.6.2 功能描述.....	311
8.6.3 API 参考	312

8.6.4 数据类型.....	314
8.7 去噪算法.....	316
8.7.1 功能描述.....	316
8.7.2 API 参考.....	316
8.7.3 数据类型.....	318
8.8 AINR.....	337
8.8.1 功能描述.....	337
8.8.2 API 参考.....	337
8.8.3 数据类型.....	346
8.9 Dehaze.....	349
8.9.1 功能描述.....	349
8.9.2 API 参考.....	349
8.9.3 数据类型.....	352
8.10 Demosaic.....	355
8.10.1 功能描述.....	355
8.10.2 API 参考.....	355
8.10.3 数据类型.....	357
8.11 黑电平.....	362
8.11.1 功能描述.....	362
8.11.2 API 参考.....	362
8.11.3 数据类型.....	364
8.12 去 FPN.....	366
8.12.1 功能描述.....	366
8.12.2 API 参考.....	367
8.12.3 数据类型.....	370
8.13 GCAC.....	372
8.13.1 功能描述.....	372

8.13.2 API 参考	373
8.13.3 数据类型.....	375
8.14 LCAC.....	377
8.14.1 功能描述.....	377
8.14.2 API 参考	377
8.14.3 数据类型.....	379
8.15 WDR.....	382
8.15.1 功能描述.....	382
8.15.2 API 参考	383
8.15.3 数据类型.....	385
8.16 LCE	393
8.16.1 功能描述.....	393
8.16.2 API 参考	393
8.16.3 数据类型.....	396
8.17 CA	405
8.17.1 功能描述.....	405
8.17.2 API 参考	406
8.17.3 数据类型.....	408
8.18 Radial Crop	411
8.18.1 功能描述.....	411
8.18.2 API 参考	411
8.18.3 数据类型.....	413
8.19 CSC.....	415
8.19.1 功能描述.....	415
8.19.2 API 参考	415
8.19.3 数据类型.....	417
8.20 CLUT	421

8.20.1 功能描述.....	421
8.20.2 API 参考	421
8.20.3 数据类型.....	423
8.21 Radial Shading.....	426
8.21.1 功能描述.....	426
8.21.2 API 参考	426
8.21.3 数据类型.....	430
8.22 RGB-IR.....	434
8.22.1 功能描述.....	434
8.22.2 API 参考	435
8.22.3 数据类型.....	437
8.23 Expander.....	440
8.23.1 功能描述.....	440
8.23.2 API 参考	441
8.23.3 数据类型.....	443
8.24 STNR.....	445
8.24.1 功能描述.....	445
8.24.2 API 参考	445
8.24.3 数据类型.....	447
8.25 Sharpen.....	482
8.25.1 功能描述.....	482
8.25.2 API 参考	482
8.25.3 数据类型.....	484
8.26 AFC.....	495
8.26.1 功能描述.....	495
8.26.2 API 参考	496
8.26.3 数据类型.....	498

9 统计信息	501
9.1 概述	501
9.2 功能描述	501
9.2.1 AE 统计信息	501
9.2.2 AWB 统计信息	503
9.3 API 参考	505
9.4 数据类型	511
10 Debug	548
10.1 概述	548
10.2 功能描述	548
10.3 API 参考	548
10.4 数据类型	552
11 错误码	556
12 Proc 调试信息说明	558
12.1 概述	558
12.2 ISP	558
13 3A 开发指南概述	583
13.1 概述	583
13.2 功能描述	583
14 3A 开发使用者指南	584
14.1 软件流程	584
14.2 Sensor 对接	584
15 开发者指南	585
15.1 概述	585
15.2 AE 算法注册 ISP 库	585
15.3 AWB 算法注册 ISP 库	587

15.4 AF 统计信息使用说明.....	588
15.4.1 概述.....	588
15.4.2 输入图像的裁剪.....	590
15.4.3 Bayer 域配置.....	590
15.4.4 抑制光源对 FV 值的影响.....	591
15.4.5 Coring	592
15.4.6 统计模式配置	593
15.4.7 统计信息配置注意事项.....	594
15.4.8 FV 值计算.....	594
16 附录.....	596
16.1 注册函数的关系	596
16.2 扩展性的设计考虑	597

插图目录

图 1-1 ISP 控制结构示意图.....	1
图 1-2 ISP firmware 架构.....	2
图 1-3 ISP firmware 内部流程.....	3
图 1-4 ISP firmware 软件结构.....	3
图 1-5 ISP firmware 使用流程.....	4
图 1-6 ISP firmware 文件组织.....	7
图 2-1 ISP 库与 sensor 库间的接口.....	19
图 2-2 SENSOR 库与 AE 库的接口.....	22
图 2-3 ISP 库与 AE 库间的接口.....	22
图 2-4 SENSOR 库和 AWB 库间的接口.....	25
图 2-5 ISP 库与 AWB 库间的接口.....	26
图 2-6 ISP 库与 AF 库间的接口.....	29
图 5-1 AE 模块工作流程图.....	129
图 5-2 AE 1024 段统计信息直方图.....	130
图 5-3 AE 工作原理图.....	131
图 5-4 AE 分配路线示意图.....	140
图 6-1 AWB 工作原理图.....	197
图 6-2 红、肤、黄、绿、蓝、紫对应的角度.....	247
图 7-1 PDAF 对焦示意图（分别为焦后与焦前）.....	251
图 8-1 动态范围压缩.....	275

图 8-2 Crosstalk 门限.....	311
图 8-3 去马赛克处理图.....	355
图 8-4 FPN 标定示意图.....	367
图 8-5 FPN 校正示意图.....	367
图 8-6 色差图解	373
图 8-7 WDR 算法合成模型	383
图 8-8 COLOR_3D_LUT 示意图.....	421
图 8-9 传统 RGB-IR 2x2 pattern 排列.....	435
图 8-10 RGB-IR 4x4 pattern 排列.....	435
图 9-1 AWB 有效像素颜色空间.....	504
图 9-2 pre_gamma 的曲线示意图.....	528
图 14-1 Sensor 适配示意图	584
图 15-1 Sensor 适配示意图	585
图 15-2 AWB 算法向 ISP 库注册的回调函数	587
图 15-3 FV 曲线.....	589
图 15-4 AF 统计模块框图.....	589
图 15-5 AF 4 组滤波器.....	590
图 15-6 AF 输入图像裁剪.....	590
图 15-7 Bayer 域统计预处理.....	591
图 15-8 关于 Gamma 配置的说明	591
图 15-9 高亮统计值.....	591
图 15-10 Level Depend Gain 曲线.....	592
图 15-11 Coring 曲线.....	593
图 15-12 滤波器后的图像.....	593

表格目录

表 9-1 AE 直方图统计信息..... 502

表 9-2 AE 直方图统计信息..... 502

表 11-1 ISP API 错误码 556

表 15-1 AE 算法向 ISP 库注册的回调函数..... 586

表 15-2 pfn_ae_ctrl 函数的 cmd、vlaue 参数解释..... 586

表 15-3 AWB 算法向 ISP 库注册的回调函数 588

表 15-4 统计信息配置注意事项..... 594

1 概述

1.1 概述

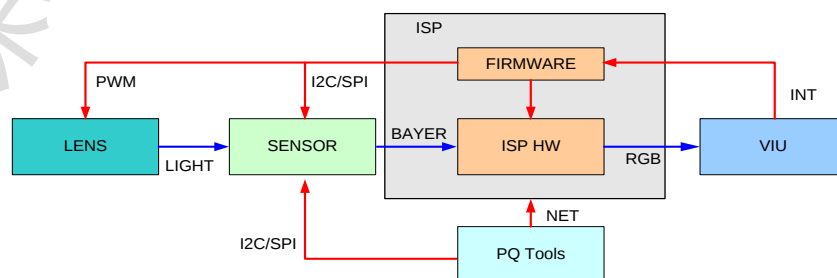
ISP 通过一系列数字图像处理算法完成对数字图像的效果处理。主要包括 3A、坏点校正、去噪、强光抑制、背光补偿、色彩增强、镜头阴影校正等处理。ISP 包括逻辑部分以及运行在其上的 firmware。这里主要介绍 ISP 的用户接口。

1.2 功能描述

ISP 的控制结构如图 1-1 所示，lens 将光信号投射到 sensor 的感光区域后，sensor 经过光电转换，将 Bayer 格式的原始图像送给 ISP，ISP 经过算法处理，输出 RGB 空间域的图像给后端的视频采集单元。在这个过程中，ISP 通过运行在其上的 firmware 对 ISP 逻辑，lens 和 sensor 进行相应控制，进而完成自动光圈、自动曝光、自动白平衡等功能。其中，firmware 的运转靠视频采集单元的中断驱动。PQ Tools 工具通过网口或者串口完成对 ISP 的在线图像质量调节。

ISP 由 ISP 逻辑及运行在其上的 Firmware 组成，逻辑单元除了完成部分算法处理外，还可以统计出当前图像的实时信息。Firmware 通过获取 ISP 逻辑的图像统计信息，重新计算，反馈控制 lens、sensor 和 ISP 逻辑，以达到自动调节图像质量的目的。

图1-1 ISP 控制结构示意图

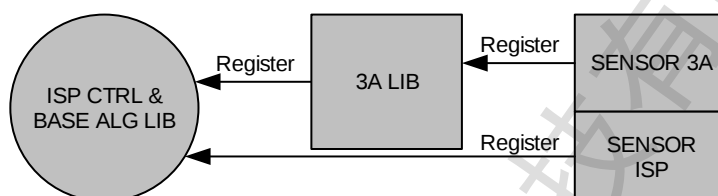


ISP 逻辑主要流程、具体概念和功能点请参见芯片手册。

1.2.1 架构

ISP 的 Firmware 包含三部分，一部分是 ISP 控制单元和基础算法库，一部分是 AE/AWB 算法库，一部分是 sensor 库。Firmware 设计的基本思想是单独提供 3A 算法库，由 ISP 控制单元调度基础算法库和 3A 算法库，同时 sensor 库分别向 ISP 基础算法库和 3A 算法库注册函数回调，以实现差异化的 sensor 适配。ISP firmware 架构如图 1-2 所示。

图1-2 ISP firmware 架构



不同的 sensor 都以回调函数的形式，向 ISP 算法库注册控制函数。ISP 控制单元调度基础算法库和 3A 算法库时，将通过这些回调函数获取初始化参数，并控制 sensor，如调节曝光时间、模拟增益、数字增益，控制 lens 步进聚焦或旋转光圈等。

1.2.2 开发模式

SDK 支持用户使用多种开发模式：

用户使用 XMEDIA 的 3A 算法库。这时用户需要根据 ISP 基础算法库和 3A 算法库给出的 sensor 适配接口去适配不同的 sensor。每款 sensor 对应一个文件夹，文件夹中包含两个主要文件：

- sensor.c

该文件中主要实现 ISP 需要的回调函数，这些回调函数中包含了 sensor 的适配算法，不同的 sensor 可能有所不同。

- sensor_ctrl.c

sensor 的底层控制驱动，主要实现 sensor 的读写和初始化动作。用户可以根据 sensor 的 datasheet 进行这两个文件的开发，必要的时候可以向 sensor 厂家寻求支持。

用户根据 ISP 库提供的 3A 算法注册接口，实现自己的 3A 算法库开发。这时用户需要根据 ISP 基础算法库和用户的 3A 算法库给出的 sensor 适配接口去适配不同的 sensor。

用户部分使用 XMEDIA_3A 算法库，部分实现自己的 3A 算法库。例如 AE 使用 libxmedia_ae.a, AWB 使用自己的 3A 算法库。SDK 提供了灵活多变的支持方式。

1.2.3 内部流程

Firmware 内部流程分两部分，如图 1-3 所示。一部分是初始化任务，主要完成 ISP 控制单元的初始化、ISP 基础算法库的初始化、3A 算法库的初始化，包括调用 sensor 的回调获取 sensor 差异化的初始化参数；另一部分是动态调节过程，在这个过程中，firmware 中的 ISP 控制单元调度 ISP 基础算法库和 3A 算法库，实时计算并进行相应控制。Firmware 的软件结构如图 1-4 所示。

图1-3 ISP firmware 内部流程

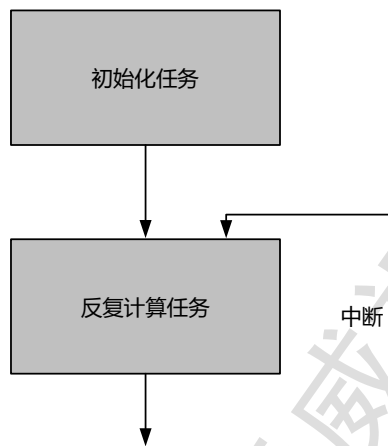
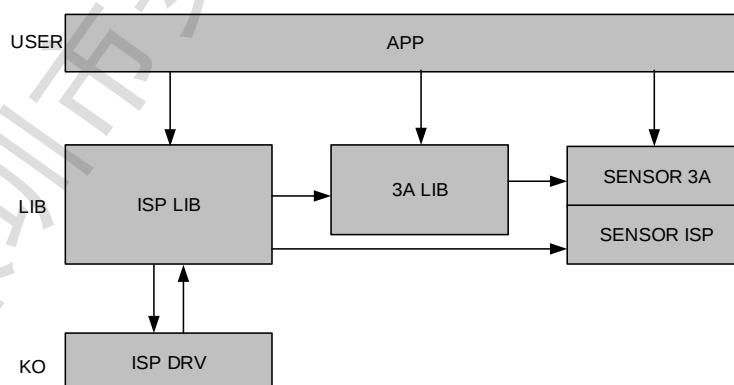


图1-4 ISP firmware 软件结构

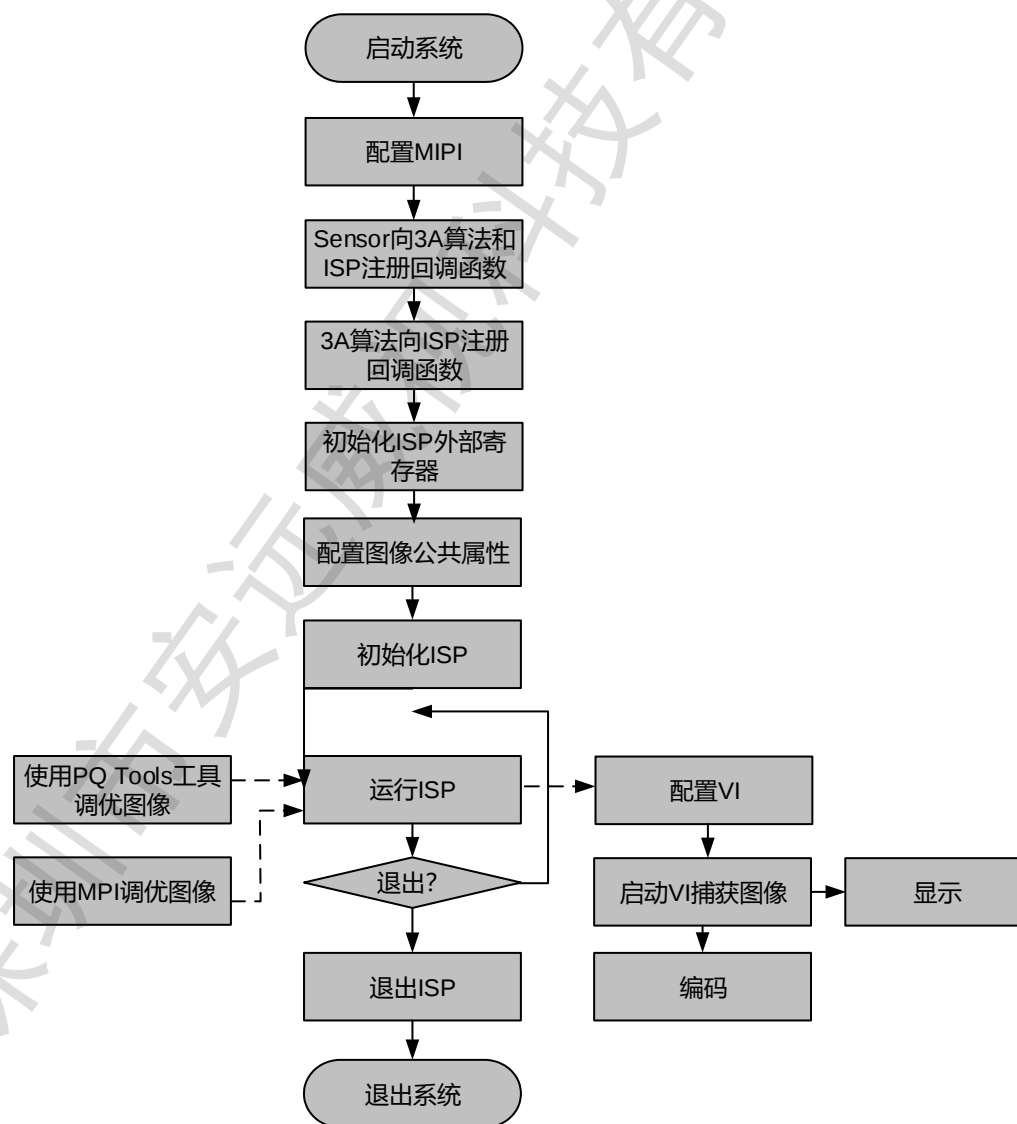


1.2.4 软件流程

ISP 作为前端采集部分，需要和视频采集单元（VIU）协同工作。ISP 初始化和基本配置完成后，需要 VIU 进行接口时序匹配。一是为了匹配不同 sensor 的输入时序，二是为 ISP 配置正确的输入时序。待时序配置完成后，ISP 就可以启动 Run 来进行动态图像质量调节。此时输出的图像被 VIU 采集，进而送去显示或编码。软件使用流程如图 1-5 所示。

PQ Tools 工具主要完成在 PC 端进行动态图像质量调节，可以调节多个影响图像质量的因子，如去噪强度、色彩转换矩阵、饱和度等。

图1-5 ISP firmware 使用流程



如果用户调试好图像效果后，可以使用 PQ Tools 工具提供的配置文件保存功能进行配置参数保存。在下次启动时系统可以使用 PQ Tools 工具提供的配置文件加载功能加载已经调节好的图像参数。

代码示例：

```
xmedia_u32 pipe = 0;
xmedia_s32 ret;
xmedia_isp_config isp_cfg;
xmedia_sensor_register_info info;
xmedia_isp_ae_register_info ae_register;
xmedia_isp_awb_register_info awb_register;
//注册 sensor 库

ret = xmedia_isp_register_sensor(pipe, &info);
if (ret != XMEDIA_SUCCESS) {
    SENSOR_ERR("sensor register failed! error code = %d.\n", ret);
    return ret;
}

//注册 XMEDIA 的 AE 算法库
strcpy(ae_register.alg_name, AE_LIB_NAME);
ae_register.alg_id = pipe;
ae_register.pfn_ae_init = ae_init;
ae_register.pfn_ae_run = ae_run;
ae_register.pfn_ae_ctrl = ae_ctrl;
ae_register.pfn_ae_exit = ae_exit;
ret = xmedia_isp_register_ae_lib(pipe, &ae_register, &(ae_ctx->sns_register));
AE_CHECK_RET_RETURN(ret, "xmedia_isp_register_ae_lib failed!");

//注册 XMEDIA 的 AWB 算法库
strcpy(awb_register.alg_name, AWB_LIB_NAME);
awb_register.alg_id = pipe;
awb_register.pfn_awb_init = awb_init;
awb_register.pfn_awb_run = awb_run;
awb_register.pfn_awb_ctrl = awb_ctrl;
awb_register.pfn_awb_exit = awb_exit;
ret = xmedia_isp_register_awb_lib(pipe, &awb_register, &(awb_ctx->st_sns_register));
if (ret != XMEDIA_SUCCESS) {
    AWB_TRACE("awb register failed!\n");
}

//初始化 ISP，且在 ISP 初始化中创建了通过中断触发的 isp_run 线程
VI_CHECK_RET(xmedia_isp_init(pipe, isp_cfg));
// isp_run 函数通过单独启动线程运行
```

```

ret = pthread_create(&ctx->pid, NULL, isp_thread, (xmedia_void *) (uintptr_t) pipe);
if (ret != 0) {
    ISP_ERR("create isp running thread failed!, error: %d, %s\r\n", ret, strerror(ret));
    ret = XMEDIA_FAILURE;
    goto PTHREAD_CREATE_FAILED;
}
//启动 VI,然后启动 ISP
VI_CHECK_RET(xmedia_isp_start(pipe));
//启动 SENSOR
VI_CHECK_RET(xmedia_sensor_start(pipe));
//启动 VO 等业务

.....

//停止 VI/VO 等业务
VI_CHECK_RET(xmedia_isp_exit(pipe));
VI_CHECK_RET(xmedia_ae_unregister(pipe));
VI_CHECK_RET(xmedia_awb_unregister(pipe));
pthread_join(isp_pid, 0);

```



说明

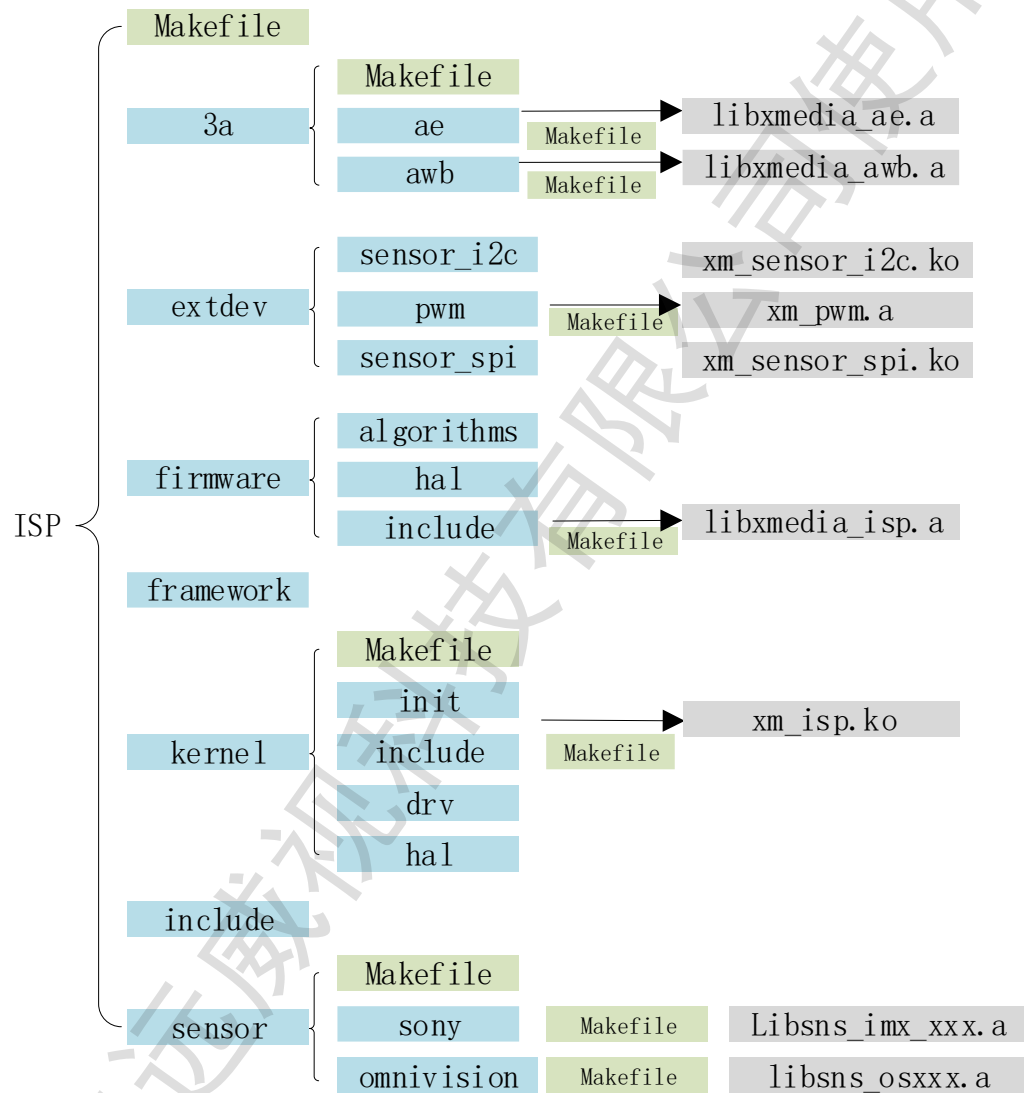
AE 库有用到标准 C 库的数学库，请使用者在 Makefile 中增加 `-lm` 编译条件。

1.2.5 文件组织

整个 ISP Firmware 的文件组织结构如图 1-6 所示，ISP 库和 3A 库、sensor 库分别独立。Firmware 中的 drv 生成的驱动程序向用户态上报 ISP 中断，并以该中断驱动 Firmware 的 ISP 控制单元运转。ISP 控制单元从驱动程序中获取统计信息，并调度基础算法单元和 3A 算法库，最后通过驱动程序配置寄存器。

图 1-6 中的 firmware 与 framework 文件夹中包含 ISP 控制单元和基础算法单元，编译后生成 `libxmedia_isp.a`，即 ISP 库。3a 文件夹中包含 AE/AWB 算法库，用户也可以基于统一的接口界面开发自己的 3a 算法。Sensor 文件夹中包含了各个 sensor 的驱动程序，该部分代码开源。

图1-6 ISP firmware 文件组织



2 系统控制

2.1 功能概述

系统控制部分包含了 ISP 公共属性配置，初始化 ISP firmware、运行 ISP firmware、退出 ISP firmware，设置 ISP 各模块等功能。

2.2 API 参考

本文档中接口，如无特殊说明，支持多进程。

- [xmedia_isp_init](#): 初始化 ISP firmware。
- [xmedia_isp_get_config](#): 获取 ISP 初始化参数。
- [xmedia_isp_exit](#): 退出 ISP firmware。
- [xmedia_isp_start](#): 启动 ISP。
- [xmedia_isp_stop](#): 停止 ISP。
- [xmedia_isp_set_attr](#): 设置 ISP 图像属性。
- [xmedia_isp_get_attr](#): 获取 ISP 图像属性。
- [xmedia_isp_freeze](#): 设置 ISP Firmware 停止或者运行 ISP Firmware。
- [xmedia_isp_export](#): 导出 ISP 算法参数。
- [xmedia_isp_import](#): 导入 ISP 算法参数。
- [xmedia_isp_register_sensor](#): ISP 提供的注册 sensor 的回调接口。
- [xmedia_isp_unregister_sensor](#): ISP 提供的反注册 sensor 的回调接口。
- [xmedia_isp_register_ae_lib](#): ISP 提供的注册 AE 库的回调接口。
- [xmedia_isp_unregister_ae_lib](#): ISP 提供的反注册 AE 库的回调接口。

- `xmedia_isp_register_awb_lib`: ISP 提供的注册 AWB 库的回调接口。
- `xmedia_isp_unregister_awb_lib`: ISP 提供的反注册 AWB 库的回调接口。
- `xmedia_isp_register_af_lib`: ISP 提供的注册 AF 库的回调接口。
- `xmedia_isp_unregister_af_lib`: ISP 提供的反注册 AF 库的回调接口。
- `xmedia_isp_set_differ_attr`: 设置多路 ISP Pipe 差异属性。
- `xmedia_isp_get_differ_attr`: 获取多路 ISP Pipe 差异属性。
- `xmedia_isp_wait_state`: 获取 ISP 中断信息。
- `xmedia_isp_set_ctrl_param`: 设置 ISP 的控制参数。
- `xmedia_isp_get_ctrl_param`: 获取 ISP 的控制参数。
- `xmedia_isp_set_smart_info`: 向 ISP 模块传递智能信息。
- `xmedia_isp_get_smart_info`: 从 ISP 模块获取智能信息。
- `xmedia_isp_get_sensor_init_param`: 获取 AE/AWB 相关默认参数。

xmedia_isp_init

【描述】

初始化 ISP firmware。

【语法】

```
xmedia_s32 xmedia_isp_init(xmedia_u32 pipe, xmedia_isp_config *cfg);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
cfg	ISP 初始化配置参数结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

该接口不支持多进程。

【相关主题】

[xmedia_isp_exit](#)

xmedia_isp_get_config

【描述】

获取 ISP 初始化参数。

【语法】

```
xmedia_s32 xmedia_isp_get_config(xmedia_u32 pipe, xmedia_isp_config *cfg);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
cfg	ISP 初始化配置参数结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败, 其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

该接口不支持多进程。

【相关主题】

无

xmedia_isp_exit

【描述】

退出 ISP firmware。

【语法】

```
xmedia_s32 xmedia_isp_exit(xmedia_u32 pipe);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

该接口不支持多进程。

【相关主题】

[xmedia_isp_init](#)

xmedia_isp_start

【描述】

启动 ISP。

【语法】

```
xmedia_s32 xmedia_isp_start(xmedia_u32 pipe);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

该接口不支持多进程。

【相关主题】

[xmedia_isp_stop](#)

xmedia_isp_stop

【描述】

停止 ISP。

【语法】

```
xmedia_s32 xmedia_isp_stop(xmedia_u32 pipe);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

该接口不支持多进程。

【相关主题】

[xmedia_isp_start](#)

xmedia_isp_set_attr

【描述】

设置 ISP 公共图像属性。

【语法】

```
xmedia_s32 xmedia_isp_set_attr(xmedia_u32 pipe, const xmedia_isp_attr *attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
attr	ISP 公共图像属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【相关主题】

[xmedia_isp_get_attr](#)

xmedia_isp_get_attr

【描述】

获取 ISP 公共图像属性。

【语法】

```
xmedia_s32 xmedia_isp_get_attr(xmedia_u32 pipe, xmedia_isp_attr *attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
attr	ISP 公共图像属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【举例】

无

【相关主题】

[xmedia_isp_set_attr](#)

xmedia_isp_freeze

【描述】

设置 ISP firmware 运行状态。

【语法】

```
xmedia_s32 xmedia_isp_freeze(xmedia_u32 pipe, xmedia_bool freeze_en);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
freeze_en	ISP Firmware 运行状态使能开关。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

在使能的情况下，isp 不更新算法参数。

【相关主题】

无

xmedia_isp_export

【描述】

导出 ISP 算法参数。

【语法】

```
xmedia_s32 xmedia_isp_export(xmedia_u32 pipe, xmedia_isp_module_info
module_info[XMEDIA_ISP_MODULE_MAX_NUM]);
```

 说明
#define XMEDIA_ISP_MODULE_MAX_NUM 4

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
module_info	算法参数结构体数组。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【相关主题】

[xmedia_isp_import](#)

xmedia_isp_import

【描述】

导入 ISP 算法参数。

【语法】

```
xmedia_s32 xmedia_isp_import(xmedia_u32 pipe, const xmedia_isp_module_info
module_info[XMEDIA_ISP_MODULE_MAX_NUM]);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
module_info	算法参数结构体数组。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【相关主题】

[xmedia_isp_export](#)

xmedia_isp_register_sensor

【描述】

ISP 提供的注册 sensor 的回调接口。

【语法】

```
xmedia_s32 xmedia_isp_register_sensor(xmedia_u32 pipe, const xmedia\_sensor\_register\_info *info);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
info	向 ISP 注册的 Sensor 的属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

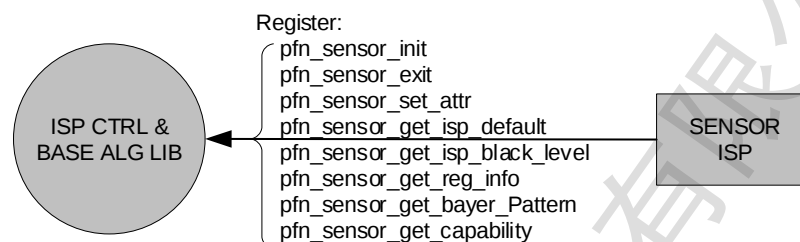
【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

- Sensor dev id 是 sensor 库中自定义的值，主要用于校对向 ISP 注册的 sensor 和向 3A 注册的 sensor 是否为同一个 sensor。
- ISP 通过 sensor 注册的一系列回调接口，获取差异化的初始化参数，并控制 sensor。
- 此接口不支持多进程操作。

图2-1 ISP 库与 sensor 库间的接口



【举例】

```
xmedia_u32 pipe = 0;
xmedia_s32 ret;
xmedia_sensor_register_info info;
memset(&info, 0, sizeof(xmedia_sensor_register_info));
info.dev_id = 0;
memcpy(info->sensor_name, SENSOR_NAME, sizeof(SENSOR_NAME));
info->isp_func.pfn_sensor_init = pfn_sensor_init;
info->isp_func.pfn_sensor_exit = pfn_sensor_exit;
info->isp_func.pfn_sensor_set_attr = pfn_sensor_set_attr;
info->isp_func.pfn_sensor_get_isp_default = pfn_sensor_get_isp_default;
info->isp_func.pfn_sensor_get_isp_black_level = pfn_sensor_get_isp_black_level;
info->isp_func.pfn_sensor_get_reg_info = pfn_sensor_get_reg_info;
info->isp_func.pfn_sensor_get_bayer_pattern = pfn_sensor_get_bayer_pattern;
info->isp_func.pfn_sensor_get_capability = pfn_sensor_get_capability;
ret = xmedia_isp_register_sensor(pipe, &info);
if (ret != XMEDIA_SUCCESS) {
    SENSOR_ERR("Sensor register failed! error code = %d.\n", ret);
    return ret;
};
```

【相关主题】

[xmedia_isp_unregister_sensor](#)

xmedia_isp_unregister_sensor

【描述】

ISP 提供的反注册 sensor 的回调接口。

【语法】

```
xmedia_s32 xmedia_isp_unregister_sensor(xmedia_u32 pipe, xmedia_u32 dev_id);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
pipe_id	向 ISP 反注册的 sensor 设备号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

- dev_id 是 sensor 库中自定义的值，主要用于校对向 ISP 反注册的 sensor 和向 3A 反注册的 sensor 是否为同一个 sensor。
- 此接口不支持多进程操作。

【举例】

```
xmedia_u32 pipe;  
ret = xmedia_isp_unregister_sensor(pipe, dev);  
if (ret != XMEDIA_SUCCESS) {  
    SENSOR_ERR("Sensor unregister function failed!\n");  
    return ret;  
}
```

```
}
```

【相关主题】

[xmedia_isp_register_sensor](#)

xmedia_isp_register_ae_lib

【描述】

ISP 提供的注册 AE 库的回调接口。

【语法】

```
xmedia_s32 xmedia_isp_register_ae_lib(xmedia_u32 pipe, const xmedia_isp_ae_register_info  
*ae_info, xmedia_sensor_register_ae_info *sensor_func);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ae_info	AE 库结构体指针。	输入
sensor_func	AE 库提供的 sensor 注册结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

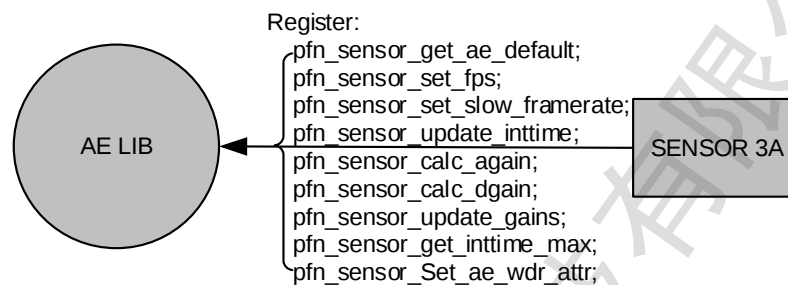
- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

- ISP 提供统一的 AE 算法库接口，初始化、运行、控制、销毁 AE 算法库。使用 XMEDIA_AE 算法库时，不需要关注此接口；使用用户自己的 AE 算法库时，需要调用此接口向 ISP 注册回调函数。

- 此接口不仅完成 AE 库向 ISP 的注册，同时完成 sensor 库向 AE 库的注册。
- 其中 sensor 库向 AE 库的注册如下图所示，通过 memcpy 函数实现注册函数的具体实现，其函数实现为此接口的子函数。
- 其中 AE 库向 ISP 库的注册如下图所示。
- 此接口不支持多进程操作。

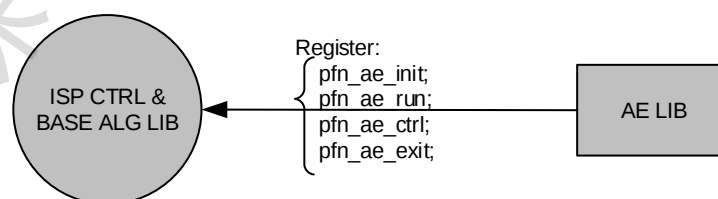
图2-2 SENSOR 库与 AE 库的接口



【举例】

```
xmedia_u32 pipe = 0;
xmedia_u32 ret;
xmedia_sensor_register_ae_info sensor_func = &(ae_ctx->sns_register);
xmedia_sensor_register_ae_func sensor_ae_func;
memset(sensor_func, 0, sizeof(xmedia_sensor_register_ae_info));
ret = isp_sensor_get_ae_callback(pipe, &sensor_ae_func); //实现 sensor_ae_func 的填充，即上图
所示的 ae 库的 sensor 结构体函数指针的填充。
if (ret == XMEDIA_SUCCESS) {
    sensor_func->dev_id = register_info->sensor_info.sns_info.dev_id;
    memcpy(&sensor_func->func, &sensor_ae_func, sizeof(xmedia_sensor_register_ae_func));
} else if (ret != XMEDIA_ERRCODE_NOT_EXIST) {
    ISP_ERR("error code = 0x%X.\n", ret);
    return ret;
}
```

图2-3 ISP 库与 AE 库间的接口



【举例】

```

xmedia_isp_ae_register_info ae_register;
xmedia_s32 ret = XMEDIA_SUCCESS;
strcpy(ae_register.alg_name, AE_LIB_NAME);
ae_register.alg_id = pipe;
ae_register.pfn_ae_init = ae_init;
ae_register.pfn_ae_run = ae_run;
ae_register.pfn_ae_ctrl = ae_ctrl;
ae_register.pfn_ae_exit = ae_exit;
ret = xmedia_isp_register_ae_lib(pipe, &ae_register, &(ae_ctx->sns_register));
AE_CHECK_RET_RETURN(ret, "xmedia_isp_register_ae_lib failed!");

```

【相关主题】

[xmedia_isp_unregister_ae_lib](#)

xmedia_isp_unregister_ae_lib

【描述】

ISP 提供的反注册 AE 库的回调接口。

【语法】

```

xmedia_s32 xmedia_isp_unregister_ae_lib(xmedia_u32 pipe,
xmedia_char name[XMEDIA_ISP_ALG_NAME_MAX_SIZE], xmedia_s32 id);

```



说明

#define XMEDIA_ISP_ALG_NAME_MAX_SIZE 20

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
name	AE 向 ISP 反注册库名称。	输入
id	AE 向 ISP 反注册库的 CHID。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

- 使用 XMEDIA_AE 算法库时, 不需要关注此接口; 使用用户自己的 AE 算法库时, 需要调用此接口向 ISP 反注册回调函数。
- 此接口不支持多进程操作。

【举例】

```
xmedia_s32 ret = XMEDIA_SUCCESS;
ret = xmedia_isp_unregister_ae_lib(pipe, lib_name, channel);
if (ret != XMEDIA_SUCCESS) {
    AE_TRACE("ae unregister failed!\n");
};
```

【相关主题】

[xmedia_isp_register_ae_lib](#)

xmedia_isp_register_awb_lib

【描述】

ISP 提供的注册 AWB 库的回调接口。

【语法】

```
xmedia_s32 xmedia_isp_register_awb_lib(xmedia_u32 pipe, const xmedia_isp_awb_register_info
*awb_info, xmedia_sensor_register_awb_info *sensor_func);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
awb_info	AWB 库结构体指针。	输入
sensor_func	AWB 库提供的 sensor 注册结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

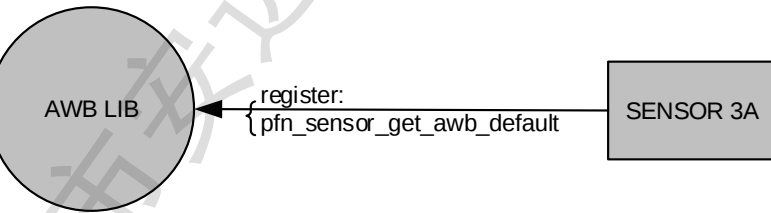
【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

- ISP 提供统一的 AWB 算法库接口，初始化、运行、控制、销毁 AWB 算法库。使用 XMEDIA_AWB 算法库时，不需要关注此接口；使用用户自己的 AWB 算法库时，需要调用此接口向 ISP 注册回调函数。
- 此接口不仅完成 AWB 库向 ISP 的注册，同时完成 sensor 库向 AWB 库的注册。
- 其中 sensor 库向 AWB 库的注册如下图所示，通过 memcpy 函数实现注册函数的具体实现，其函数实现为此接口的子函数。
- 其中 AWB 库向 ISP 库的注册如下图所示。
- 此接口不支持多进程操作。

图2-4 SENSOR 库和 AWB 库间的接口



【举例】

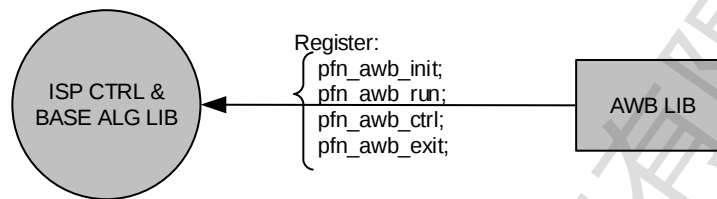
```
xmedia_s32 pipe =0;
xmedia_u32 ret;
xmedia_sensor_register_awb_info sensor_func = &(awb_ctx->sns_register);
xmedia_sensor_register_awb_func sensor_awb_func;
memset(sensor_func, 0, sizeof(xmedia_sensor_register_awb_info));
ret = isp_sensor_get_awb_callback(pipe, &sensor_awb_func);//实现 sensor_awb_func 的填充，即
上图所示的 awb 库的 sensor 结构体函数指针的填充。
```

```

if (ret == XMEDIA_SUCCESS) {
    sensor_func->dev_id = register_info->sensor_info.sns_info.dev_id;
    memcpy(&sensor_func->func, &sensor_awb_func, sizeof(xmedia_sensor_register_awb_func));
} else if (ret != XMEDIA_ERRCODE_NOT_EXIST) {
    ISP_ERR("error code = 0x%X.\n", ret);
    return ret;
};

```

图2-5 ISP 库与 AWB 库间的接口



【举例】

```

xmedia_isp_awb_register_info awb_register;
xmedia_s32 ret = XMEDIA_SUCCESS;
strcpy(awb_register.alg_name, AWB_LIB_NAME);
awb_register.alg_id = pipe;
awb_register.pfn_awb_init = awb_init;
awb_register.pfn_awb_run = awb_run;
awb_register.pfn_awb_ctrl = awb_ctrl;
awb_register.pfn_awb_exit = awb_exit;
ret = xmedia_isp_register_awb_lib(pipe, &awb_register, &(awb_ctx->sns_register));
if (ret != XMEDIA_SUCCESS) {
    AWB_TRACE("awb register failed!\n");
};

```

【相关主题】

[xmedia_isp_unregister_awb_lib](#)

xmedia_isp_unregister_awb_lib

【描述】

ISP 提供的反注册 AWB 库的回调接口。

【语法】

```

xmedia_s32 xmedia_isp_unregister_awb_lib(xmedia_u32 pipe, xmedia_char
name[XMEDIA_ISP_ALG_NAME_MAX_SIZE], xmedia_s32 id);

```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
name	AWB 库算法名称。	输入
id	AWB 库算法 id 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【举例】

- 使用 XMEDIA_AWB 算法库时，不需要关注此接口；使用用户自己的 AWB 算法库时，需要调用此接口向 ISP 反注册回调函数。
- 此接口不支持多进程操作。

【相关主题】

[xmedia_isp_register_awb_lib](#)

xmedia_isp_register_af_lib

【描述】

ISP 提供的注册 AF 库的回调接口。

【语法】

```
xmedia_s32 xmedia_isp_register_af_lib(xmedia_u32 pipe, const xmedia_isp_af_register_info
*af_info, xmedia_lens_register_af_info *lens_func);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
af_info	AF 库结构体指针。	输入
lens_func	AF 库提供的 lens 注册结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

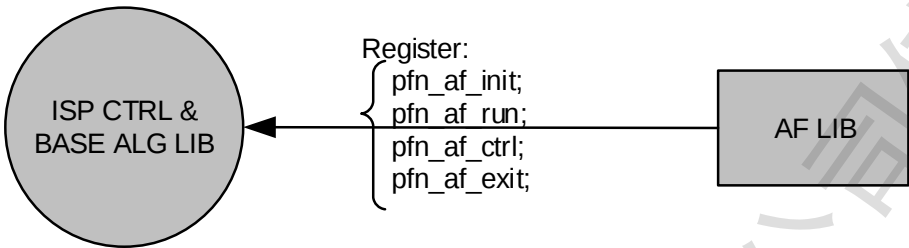
【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

- ISP 提供统一的 AF 算法库接口，初始化、运行、控制、销毁 AF 算法库。使用 XMEDIA_AF 算法库时，不需要关注此接口；使用用户自己的 AF 算法库时，需要调用此接口向 ISP 注册回调函数。
- 此接口不仅完成 AF 库向 ISP 的注册，同时完成 sensor 库向 AF 库的注册,但目前没有具体的 sensor 库向 AF 注册的函数实现，只是保留其接口。
- 此接口不支持多进程操作。

图2-6 ISP 库与 AF 库间的接口



【相关主题】

[xmedia_isp_unregister_af_lib](#)

xmedia_isp_unregister_af_lib

【描述】

ISP 提供的反注册 AF 库的回调接口。

【语法】

```
xmedia_s32 xmedia_isp_unregister_af_lib(xmedia_u32 pipe,
xmedia_char name[XMEDIA_ISP_ALG_NAME_MAX_SIZE], xmedia_s32 id);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
name	AF 向 ISP 反注册库名称。	输入
id	AF 向 ISP 反注册库的 CHID	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

- 使用 XMEDIA_AF 算法库时, 不需要关注此接口; 使用用户自己的 AF 算法库时, 需要调用此接口向 ISP 反注册回调函数。
- 此接口不支持多进程操作。

【相关主题】

[xmedia_isp_register_af_lib](#)

xmedia_isp_set_differ_attr

【描述】

设置多路 ISP Pipe 差异属性。

【语法】

```
xmedia_s32 xmedia_isp_set_differ_attr(xmedia_u32 pipe, const xmedia_isp_diff_attr *differ);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
differ	多路 ISP 差异属性结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败, 其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【相关主题】

[xmedia_isp_get_differ_attr](#)

xmedia_isp_get_differ_attr

【描述】

获取多路 ISP Pipe 差异属性。

【语法】

```
xmedia_s32 xmedia_isp_get_differ_attr(xmedia_u32 pipe, const xmedia_isp_diff_attr *differ);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
differ	多路 ISP 差异属性结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_isp.h
- 库文件：libxmedia_isp.a

【注意】

无

【相关主题】

[xmedia_isp_set_differ_attr](#)

xmedia_isp_set_ctrl_param

【描述】

设置 ISP 控制参数。

【语法】

```
xmedia_s32 xmedia_isp_set_ctrl_param(xmedia_u32 pipe, const xmedia_isp_ctrl_param
*isp_ctrl_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
isp_ctrl_attr	ISP 控制参数结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【相关主题】

[xmedia_isp_get_ctrl_param](#)

xmedia_isp_get_ctrl_param

【描述】

获取 ISP 控制参数。

【语法】

```
xmedia_s32 xmedia_isp_get_ctrl_param(xmedia_u32 pipe, xmedia\_isp\_ctrl\_param *isp_ctrl_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
isp_ctrl_attr	ISP 控制参数结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【相关主题】

[xmedia_isp_set_ctrl_param](#)

xmedia_isp_wait_state

【描述】

获取 ISP 中断信息。

【语法】

```
xmedia_s32 xmedia_isp_wait_state(xmedia_u32 pipe, xmedia\_isp\_wait\_type wait_type,  
xmedia_u32 millisec);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
wait_type	等待中断信号类型。	输入
millisec	超时时间，单位 ms。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_isp.h
- 库文件：libxmedia_isp.a

【注意】

无

【相关主题】

无

xmedia_isp_set_smart_info

【描述】

向 ISP 模块传递智能信息。

【语法】

```
xmedia_s32 xmedia_isp_set_smart_info(xmedia_u32 pipe, const xmedia_isp_smart_info
*smart_info);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入。
smart_info	智能信息，包括人脸、人形信息、隧道检测信息。	输入。

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无。

【相关主题】

[xmedia_isp_get_smart_info](#)

xmedia_isp_get_smart_info

【描述】

从 ISP 模块获取智能信息。

【语法】

```
xmedia_s32 xmedia_isp_get_smart_info(xmedia_u32 pipe, xmedia\_isp\_smart\_info *smart_info);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
smart_info	智能信息，包括人脸、人形信息、隧道检测信息。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无。

【相关主题】

[xmedia_isp_set_smart_info](#)

xmedia_isp_get_sensor_init_param

【描述】

获取 AE/AWB 相关默认参数。

【语法】

```
xmedia_s32 xmedia_isp_get_sensor_init_param(xmedia_u32 pipe, xmedia_sensor_init_param
*init_param);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
init_param	AE/AWB 初始化参数结构体。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【相关主题】

[xmedia_sensor_init_param](#)

2.3 数据类型

本文档中变量，如未明确指定取值范围，则默认是数据类型对应的取值范围。例如 xmedia_u8 数据类型的变量取值范围为[0, 255]。本文档中变量，如未明确指定数据精度，则默认是 1。

- [xmedia_video_size](#): 定义图像宽高。
- [xmedia_video_wdr_mode](#): 定义 wdr 模式。
- [xmedia_video_pixel_format](#): 定义图像的像素格式。
- [xmedia_video_bayer_format](#): 定义 Bayer 数据格式。
- [xmedia_isp_work_mode](#): 定义 ISP 工作模式。
- [xmedia_isp_mode_config](#): 定义 ISP 工作模式配置。

- `xmedia_isp_config`: 定义 ISP 初始化配置信息。
- `xmedia_isp_module_info`: 定义算法模块信息。
- `xmedia_isp_attr`: ISP 图像属性。
- `xmedia_sensor_register_isp_func`: 定义 ISP 回调函数结构体信息。
- `xmedia_sensor_register_awb_func`: 定义 AWB 回调函数结构体信息。
- `xmedia_sensor_register_ae_func`: 定义 AE 回调函数结构体信息。
- `xmedia_sensor_register_info`: 定义 sensor 注册信息。
- `xmedia_isp_ae_register_info`: 定义 AE 库向 ISP 的注册信息。
- `xmedia_isp_ae_info`: 定义 ae 库 run 函数需要的统计信息。
- `xmedia_isp_ae_result`: 定义 ae 库 run 函数输出的配置信息。
- `xmedia_sensor_register_ae_info`: 定义 sensor 向 AE 的注册信息。
- `xmedia_isp_awb_register_info`: 定义 AWB 库向 ISP 库的注册信息。
- `xmedia_isp_awb_info`: 定义 isp 提供给 AWB 的统计信息结构体。
- `xmedia_isp_awb_result`: 定义 awb 输出结果。
- `xmedia_isp_awb_raw_stat_attr`: 定义 AWB 库返回给 ISP 的配置寄存器结构体。
- `xmedia_sensor_register_awb_info`: 定义 sensor 向 AWB 库的注册信息结构体。
- `xmedia_sensor_register_af_info`: 定义 sensor 向 AF 库的注册信息。
- `xmedia_isp_af_register_info`: 定义 AF 库向 ISP 库的注册信息。
- `xmedia_isp_af_param`: 定义 ISP 提供给 AF 库的初始化参数。
- `xmedia_isp_af_result`: 定义 AF 库返给 ISP 的参数。
- `xmedia_isp_af_info`: 定义 ISP 提供给 AF 库的统计信息相关参数。
- `xmedia_isp_af_pdaf_statistics`: 定义 ISP 提供给 AF 库的 pdaf 统计信息相关参数。
- `xmedia_isp_fe_af_zone_matrix`: 定义 ISP 提供给 AF 库的 cdaf fe 统计信息相关参数。
- `xmedia_isp_be_af_zone_matrix`: 定义 ISP 提供给 AF 库的 cdaf fe 统计信息相关参数。
- `xmedia_lens_register_info`: 定义 LENS 向 ISP 注册的信息。
- `xmedia_lens_register_isp_func`: 定义 LENS 向 ISP 注册的回调函数。
- `xmedia_lens_register_ae_info`: 定义 LENS 向 AE 库的注册信息。
- `xmedia_lens_register_ae_func`: 定义 LENS 向 AE 库注册的回调函数。
- `xmedia_lens_register_awb_info`: 定义 LENS 向 AWB 库的注册信息。

- `xmedia_lens_register_af_info`: 定义 LENS 向 AF 库的注册信息。
- `xmedia_lens_register_af_func`: 定义 LENS 向 AF 库注册的回调函数。
- `xmedia_isp_diff_attr`: 定义多路 ISP 差异属性信息。
- `xmedia_isp_wait_type`: 定义 ISP 中断类型。
- `xmedia_isp_ctrl_param`: 定义 ISP 控制信息。
- `xmedia_isp_iris_f_no`: 定义 ISP 光圈 F 值。
- `xmedia_isp_smart_roi`: 定义 ISP 提供给 AE 的人形、人脸统计信息结构体。
- `xmedia_isp_smart_info`: 定义 ISP 智能信息。

`xmedia_video_size`

【说明】

定义图像宽高。

【定义】

```
typedef struct {
    xmedia_u32 width;
    xmedia_u32 height;
} xmedia_video_size;
```

【成员】

成员名称	描述
width	图像宽度。取值范围: [1, 8192]
height	图像高度。取值范围: [1, 8192]

【注意事项】

无

【相关数据类型及接口】

无

`xmedia_video_wdr_mode`

【说明】

定义 wdr 模式。

【定义】

```
typedef enum {  
    XMEDIA_VIDEO_WDR_MODE_NONE = 0x0,  
    XMEDIA_VIDEO_WDR_MODE_BUILT_IN,  
    XMEDIA_VIDEO_WDR_MODE_2TO1_LINE,  
    XMEDIA_VIDEO_WDR_MODE_2TO1_FRAME,  
    XMEDIA_VIDEO_WDR_MODE_3TO1_LINE,  
    XMEDIA_VIDEO_WDR_MODE_3TO1_FRAME,  
    XMEDIA_VIDEO_WDR_MODE_4TO1_LINE,  
    XMEDIA_VIDEO_WDR_MODE_4TO1_FRAME,  
    XMEDIA_VIDEO_WDR_MODE_MAX,  
} xmedia_video_wdr_mode;
```

【成员】

成员名称	描述
XMEDIA_VIDEO_WDR_MODE_NONE	线性模式。
XMEDIA_VIDEO_WDR_MODE_BUILT_IN	Sensor 合成 WDR 模式。
XMEDIA_VIDEO_WDR_MODE_2TO1_LINE	2 帧合成行 WDR 模式。
XMEDIA_VIDEO_WDR_MODE_2TO1_FRAME	2 帧合成帧 WDR 模式。
XMEDIA_VIDEO_WDR_MODE_3TO1_LINE	3 帧合成行 WDR 模式。
XMEDIA_VIDEO_WDR_MODE_3TO1_FRAME	3 帧合成帧 WDR 模式。
XMEDIA_VIDEO_WDR_MODE_4TO1_LINE	4 帧合成行 WDR 模式。
XMEDIA_VIDEO_WDR_MODE_4TO1_FRAME	4 帧合成帧 WDR 模式。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_video_pixel_format

【说明】

定义图像的像素格式。

【定义】

```
typedef enum {  
    /* Semi-Planner */  
    XMEDIA_VIDEO_PIXEL_FMT_YUV_SEMIPLANAR_420 = 0,  
    XMEDIA_VIDEO_PIXEL_FMT_YVU_SEMIPLANAR_420,  
    XMEDIA_VIDEO_PIXEL_FMT_YUV_SEMIPLANAR_422,  
    XMEDIA_VIDEO_PIXEL_FMT_YVU_SEMIPLANAR_422,  
    XMEDIA_VIDEO_PIXEL_FMT_YUV_SEMIPLANAR_444,  
    XMEDIA_VIDEO_PIXEL_FMT_YVU_SEMIPLANAR_444,  
    /* Package */  
    XMEDIA_VIDEO_PIXEL_FMT_YUYV_PACKAGE_422 = 0x40,  
    XMEDIA_VIDEO_PIXEL_FMT_YVYU_PACKAGE_422,  
    XMEDIA_VIDEO_PIXEL_FMT_UYVY_PACKAGE_422,  
    XMEDIA_VIDEO_PIXEL_FMT_VYUY_PACKAGE_422,  
    XMEDIA_VIDEO_PIXEL_FMT_YYUV_PACKAGE_422,  
    XMEDIA_VIDEO_PIXEL_FMT_YYVU_PACKAGE_422,  
    XMEDIA_VIDEO_PIXEL_FMT_UVYY_PACKAGE_422,  
    XMEDIA_VIDEO_PIXEL_FMT_VUYY_PACKAGE_422,  
    XMEDIA_VIDEO_PIXEL_FMT_VY1UY0_PACKAGE_422,  
    /* yuv 400 */  
    XMEDIA_VIDEO_PIXEL_FMT_YUV_400 = 0x90,  
    /* RGB */  
    XMEDIA_VIDEO_PIXEL_FMT_ARGB_1555 = 0xC0,  
    XMEDIA_VIDEO_PIXEL_FMT_ARGB_4444,  
    XMEDIA_VIDEO_PIXEL_FMT_ARGB_8888,  
    XMEDIA_VIDEO_PIXEL_FMT_ABGR_1555,  
    XMEDIA_VIDEO_PIXEL_FMT_ABGR_4444,  
    XMEDIA_VIDEO_PIXEL_FMT_ABGR_8888,  
    XMEDIA_VIDEO_PIXEL_FMT_RGBA_1555,  
    XMEDIA_VIDEO_PIXEL_FMT_RGBA_4444,  
    XMEDIA_VIDEO_PIXEL_FMT_RGBA_8888,  
    XMEDIA_VIDEO_PIXEL_FMT_BGRA_1555,  
    XMEDIA_VIDEO_PIXEL_FMT_BGRA_4444,  
    XMEDIA_VIDEO_PIXEL_FMT_BGRA_8888,  
    XMEDIA_VIDEO_PIXEL_FMT_RGB_565,  
    XMEDIA_VIDEO_PIXEL_FMT_RGB_888,  
    XMEDIA_VIDEO_PIXEL_FMT_BGR_565,  
    XMEDIA_VIDEO_PIXEL_FMT_BGR_888,  
    /* CLUT */  
    XMEDIA_VIDEO_PIXEL_FMT_CLUT2 = 0x100,  
    XMEDIA_VIDEO_PIXEL_FMT_CLUT4,  
    XMEDIA_VIDEO_PIXEL_FMT_CLUT8,
```

```

/*RAW*/
XMEDIA_VIDEO_PIXEL_FMT_RAW,
XMEDIA_VIDEO_PIXEL_FMT_MAX = 0x120
} xmedia_video_pixel_format;

```

【成员】

成员名称	描述
XMEDIA_VIDEO_PIXEL_FMT_YUV_SEMIPLANAR_420	YUV420
XMEDIA_VIDEO_PIXEL_FMT_YVU_SEMIPLANAR_420	YVU420
XMEDIA_VIDEO_PIXEL_FMT_YUV_SEMIPLANAR_422	YUV422
XMEDIA_VIDEO_PIXEL_FMT_YVU_SEMIPLANAR_422	YVU422
XMEDIA_VIDEO_PIXEL_FMT_YUV_SEMIPLANAR_444	YUV444
XMEDIA_VIDEO_PIXEL_FMT_YVU_SEMIPLANAR_444	YVU444
XMEDIA_VIDEO_PIXEL_FMT_YUYV_PACKAGE_422	YUYV422
XMEDIA_VIDEO_PIXEL_FMT_YVYU_PACKAGE_422	YVYU422
XMEDIA_VIDEO_PIXEL_FMT_UYVY_PACKAGE_422	UYVY422
XMEDIA_VIDEO_PIXEL_FMT_VYUY_PACKAGE_422	VYUY422
XMEDIA_VIDEO_PIXEL_FMT_YYUV_PACKAGE_422	YYUV422
XMEDIA_VIDEO_PIXEL_FMT_YYVU_PACKAGE_422	YYVU422
XMEDIA_VIDEO_PIXEL_FMT_UVYY_PACKAGE_422	UVYY422
XMEDIA_VIDEO_PIXEL_FMT_VUYY_PACKAGE_422	VUYY422
XMEDIA_VIDEO_PIXEL_FMT_VY1UY0_PACKAGE_422	VY1YY0422
XMEDIA_VIDEO_PIXEL_FMT_YUV_400	YUV400
XMEDIA_VIDEO_PIXEL_FMT_ARGB_1555	RGB1555
XMEDIA_VIDEO_PIXEL_FMT_ARGB_4444	RGB4444
XMEDIA_VIDEO_PIXEL_FMT_ARGB_8888	RGB8888
XMEDIA_VIDEO_PIXEL_FMT_ABGR_1555	BGR1555
XMEDIA_VIDEO_PIXEL_FMT_ABGR_4444	BGR4444
XMEDIA_VIDEO_PIXEL_FMT_ABGR_8888	BGR8888
XMEDIA_VIDEO_PIXEL_FMT_RGBA_1555	RGBA1555
XMEDIA_VIDEO_PIXEL_FMT_RGBA_4444	RGBA4444
XMEDIA_VIDEO_PIXEL_FMT_RGBA_8888	RGBA8888
XMEDIA_VIDEO_PIXEL_FMT_BGRA_1555	BGRA1555

成员名称	描述
XMEDIA_VIDEO_PIXEL_FMT_BGRA_4444	BGRA4444
XMEDIA_VIDEO_PIXEL_FMT_BGRA_8888	BGRA8888
XMEDIA_VIDEO_PIXEL_FMT_RGB_565	RGB565
XMEDIA_VIDEO_PIXEL_FMT_RGB_888	RGB888
XMEDIA_VIDEO_PIXEL_FMT_BGR_565	BGR565
XMEDIA_VIDEO_PIXEL_FMT_BGR_888	BGR888
XMEDIA_VIDEO_PIXEL_FMT_CLUT2	CLUT2
XMEDIA_VIDEO_PIXEL_FMT_CLUT4	CLUT4
XMEDIA_VIDEO_PIXEL_FMT_CLUT8	CLUT8
XMEDIA_VIDEO_PIXEL_FMT_RAW	RAW

【注意事项】

无

【相关数据类型及接口】

无

xmedia_video_bayer_format

【说明】

定义 Bayer 数据格式。

【定义】

```
typedef enum {
    XMEDIA_VIDEO_BAYER_FMT_RGGB = 0,
    XMEDIA_VIDEO_BAYER_FMT_GRBG,
    XMEDIA_VIDEO_BAYER_FMT_GBRG,
    XMEDIA_VIDEO_BAYER_FMT_BGGR,
    XMEDIA_VIDEO_BAYER_FMT_RCCB,
    XMEDIA_VIDEO_BAYER_FMT_RCCG,
    XMEDIA_VIDEO_BAYER_FMT_RCCC,
    XMEDIA_VIDEO_BAYER_FMT_CCCC,
    XMEDIA_VIDEO_BAYER_FMT_RYYC,
    XMEDIA_VIDEO_BAYER_FMT_RYYB,
    XMEDIA_VIDEO_BAYER_FMT_RGBIR_2X2,
    XMEDIA_VIDEO_BAYER_FMT_RGBIR_4X4,
    XMEDIA_VIDEO_BAYER_MAX
}
```

```
} xmedia_video_bayer_format;
```

【成员】

成员名称	描述
XMEDIA_VIDEO_BAYER_FMT_RGGB	RGGB 排列方式。
XMEDIA_VIDEO_BAYER_FMT_GRBG	GRBG 排列方式。
XMEDIA_VIDEO_BAYER_FMT_GBRG	GBRG 排列方式。
XMEDIA_VIDEO_BAYER_FMT_BGGR	BGGR 排列方式。
XMEDIA_VIDEO_BAYER_FMT_RCCB	RCCB 排列方式。
XMEDIA_VIDEO_BAYER_FMT_RCCG	RCCG 排列方式。
XMEDIA_VIDEO_BAYER_FMT_RCCC	RCCC 排列方式。
XMEDIA_VIDEO_BAYER_FMT_CCCC	CCCC 排列方式。
XMEDIA_VIDEO_BAYER_FMT_RYYC	RYYC 排列方式。
XMEDIA_VIDEO_BAYER_FMT_RYYB	RYYB 排列方式。
XMEDIA_VIDEO_BAYER_FMT_RGBIR_2X2	RGBIR_2X2 排列方式。
XMEDIA_VIDEO_BAYER_FMT_RGBIR_4X4	RGBIR_4X4 排列方式。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_work_mode

【说明】

定义 ISP 工作模式。

【定义】

```
typedef enum {  
    XMEDIA_ISP_WORK_MODE_MASTER = 0,
```

```
XMEDIA_ISP_WORK_MODE_SLAVE,  
XMEDIA_ISP_WORK_MODE_MAX  
} xmedia_isp_work_mode;
```

【成员】

成员名称	描述
XMEDIA_ISP_WORK_MODE_MASTER	ISP 主模式，算法模块独立运行。
XMEDIA_ISP_WORK_MODE_SLAVE	ISP 从模式，部分或全部 ISP 参数跟随指定的主模式 PIPE。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_mode_config

【说明】

定义 ISP 工作模式配置。

【定义】

```
typedef struct {  
    xmedia_isp_work_mode work_mode;  
  
    union {  
        struct {  
            xmedia_u32 slave_num;  
            xmedia_u32 slave_pipe[XMEDIA_ISP_SLAVE_PIPE_MAX_NUM];  
            xmedia_bool blend_stat_enable;  
            xmedia_bool blend_luma_enable;  
        } master_mode;  
        struct {  
            xmedia_u32 master_pipe;  
            xmedia_u64 follow_info;  
        } slave_mode;  
    };  
} xmedia_isp_mode_config;
```

**【成员】**

成员名称	描述
work_mode	ISP 工作模式。
slave_num	从属于该主 pipe 的有效的从 pipe 个数。
slave_pipe	从属于该主 pipe 的从 pipe 号。
blend_stat_enable	统计信息融合的使能。
blend_luma_enable	luma 融合的使能。
master_mode	主模式相关配置。
master_pipe	从模式下，与该从 pipe 绑定的主 pipe 号。
follow_info	从模式下，从 pipe 需要跟随主 pipe 的算法配置，一个 bit 位代表一种算法。
slave_mode	从模式相关配置。

【注意事项】

- follow_info 的不同位代表需要 follow 的算法类型，其具体含义如下文所示，用户需要从 pipe 某种算法跟随主 pipe，就把相应位使能即可。
- 如果需要从 pipe 完全 follow 主 pipe，直接配置 0xFFFFFFFF。

```
#define XMEDIA_ISP_FOLLOW_AAC      (1 << 0)
#define XMEDIA_ISP_FOLLOW_AE      (1 << 1)
#define XMEDIA_ISP_FOLLOW_AF      (1 << 2)
#define XMEDIA_ISP_FOLLOW_AWB     (1 << 3)
#define XMEDIA_ISP_FOLLOW_BLC     (1 << 4)
#define XMEDIA_ISP_FOLLOW_BNR     (1 << 5)
#define XMEDIA_ISP_FOLLOW_CCM     (1 << 6)
#define XMEDIA_ISP_FOLLOW_CDS     (1 << 7)
#define XMEDIA_ISP_FOLLOW_CLUT    (1 << 8)
#define XMEDIA_ISP_FOLLOW_COMPRESS (1 << 9)
```

```

#define XMEDIA_ISP_FOLLOW_CROSSTALK (1 << 10)
#define XMEDIA_ISP_FOLLOW_CSC      (1 << 11)
#define XMEDIA_ISP_FOLLOW_CUS      (1 << 12)
#define XMEDIA_ISP_FOLLOW_DEHAZE   (1 << 13)
#define XMEDIA_ISP_FOLLOW_DGN      (1 << 14)
#define XMEDIA_ISP_FOLLOW_DMS      (1 << 15)
#define XMEDIA_ISP_FOLLOW_DPC      (1 << 16)
#define XMEDIA_ISP_FOLLOW_DRC      (1 << 17)
#define XMEDIA_ISP_FOLLOW_EXPANDER (1 << 18)
#define XMEDIA_ISP_FOLLOW_FPN      (1 << 19)
#define XMEDIA_ISP_FOLLOW_GAMMA    (1 << 20)
#define XMEDIA_ISP_FOLLOW_GCAC     (1 << 21)
#define XMEDIA_ISP_FOLLOW_HLC      (1 << 22)
#define XMEDIA_ISP_FOLLOW_LCE      (1 << 23)
#define XMEDIA_ISP_FOLLOW_MLSC     (1 << 24)
#define XMEDIA_ISP_FOLLOW_RC       (1 << 25)
#define XMEDIA_ISP_FOLLOW_RGBIR    (1 << 26)
#define XMEDIA_ISP_FOLLOW_RLSC     (1 << 27)
#define XMEDIA_ISP_FOLLOW_SHARPEN  (1 << 28)
#define XMEDIA_ISP_FOLLOW_WDR      (1 << 29)
#define XMEDIA_ISP_FOLLOW_CA       (1 << 30)
#define XMEDIA_ISP_FOLLOW_SENSOR   (1 << 31)
#define XMEDIA_ISP_FOLLOW_ALL      0xFFFFFFFF

```

【相关数据类型及接口】

无

xmedia_isp_config

【说明】

定义 ISP 初始化配置信息。

【定义】

```

typedef struct {
    xmedia_video_size  size;
    xmedia_float       fps;
}

```

```

xmedia_video_wdr_mode wdr_mode;
xmedia_video_pixel_format pixel_fmt;
xmedia_video_bayer_format bayer_fmt;
xmedia_isp_mode_config mode_config;
} xmedia_isp_config;

```

【成员】

成员名称	描述
size	sensor 输出图像宽高。
fps	帧率。
wdr_mode	WDR 工作模式。
pixel_fmt	像素格式。
bayer_fmt	bayer 数据格式。
mode_config	ISP 工作模式配置。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_module_info

【说明】

定义算法模块信息。

【定义】

```

typedef struct {
    xmedia_u32 module;
    xmedia_u32 version;
    xmedia_u32 length;
    xmedia_char name[XMEDIA_ISP_VREG_NAME_MAX_SIZE];
    xmedia_void *addr;
} xmedia_isp_module_info;

```

【成员】

成员名称	描述
module	模块 id，以区分 ISP/AE/AWB/AF。
version	算法 vreg 版本号，当算法参数有变动时，版本号会改变。
length	vreg 长度，单位：字节。
name	模块名称，比如： 'ISP' 'AE' 'AWB' 'AF' 。
addr	导出的 vreg 首地址，导入时是 buffer 地址。

【注意事项】

```
typedef enum {
    XMEDIA_ISP_MODULE_ID_MIN = 0x00584D00,
    XMEDIA_ISP_MODULE_ID_ISP = XMEDIA_ISP_MODULE_ID_MIN,
    XMEDIA_ISP_MODULE_ID_AWB,
    XMEDIA_ISP_MODULE_ID_AE,
    XMEDIA_ISP_MODULE_ID_AF,
    XMEDIA_ISP_MODULE_ID_MAX
} xmedia_isp_module_id;
```

模块 id 的取值需使用头文件中指定的枚举来进行赋值，如上文中所定义的枚举。

【相关数据类型及接口】

无

xmedia_isp_attr

【说明】

定义图像属性信息。

【定义】

```
typedef struct {
    xmedia_video_size      size;
    xmedia_float           fps;
    xmedia_video_wdr_mode  wdr_mode;
    xmedia_video_bayer_format bayer_fmt;
```

```
} xmedia_isp_attr;
```

【成员】

成员名称	描述
size	图像宽高信息。
fps	帧率，取值需大于 0。
wdr_mode	WDR 工作模式。
bayer_fmt	bayer 数据格式。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_sensor_register_isp_func

【说明】

定义 ISP 回调函数结构体信息。

【定义】

```
typedef struct {  
    xmedia_s32 (*pfn_sensor_get_isp_default)(xmedia_u32 dev, xmedia_sensor_isp_default  
*isp_def);  
    xmedia_s32 (*pfn_sensor_get_isp_black_level)(xmedia_u32 dev, xmedia_sensor_black_level  
*black_level);  
    xmedia_s32 (*pfn_sensor_get_reg_info)(xmedia_u32 dev, xmedia_sensor_regs_info  
*sns_regs_info);  
    xmedia_s32 (*pfn_sensor_get_bayer_pattern)(xmedia_u32 dev, xmedia_video_bayer_format  
*bayer_pattern);  
    xmedia_s32 (*pfn_sensor_get_capability)(xmedia_sensor_capability *capability);  
    xmedia_s32 (*pfn_sensor_get_fps)(xmedia_u32 dev, xmedia_float *fps, xmedia_float *min_fps,  
xmedia_float *max_fps);  
    xmedia_s32 (*pfn_sensor_get_hmax_vmax)(xmedia_u32 dev, xmedia_u32 *hmax,  
xmedia_u32*vmax);  
    xmedia_s32 (*pfn_sensor_get_trig_attr)(xmedia_sensor_trig_attr *trig_attr);  
    xmedia_s32 (*pfn_sensor_get_dev_addr)(xmedia_u32 dev, xmedia_u32 *slave_addr);  
};
```

```

xmedia_s32 (*pfn_sensor_update_reg_info)(xmedia_u32 dev, xmedia_sensor_regs_info
*cur_info, xmedia_sensor_regs_info *pre_info);
xmedia_s32 (*pfn_sensor_get_pdaf_info)(xmedia_u32 dev, xmedia_sensor_pdaf_info
*pdaf_info);
xmedia_s32 (*pfn_sensor_format_pdaf)(xmedia_u32 dev, xmedia_void *pd_buff, xmedia_void
*tmp_buff, xmedia_video_size *size, xmedia_u8 bit_width);
xmedia_s32 (*pfn_sensor_calc_image_size)(xmedia_u32 dev, const xmedia_video_size
*pd_size, xmedia_video_size *image_size);

xmedia_s32 (*pfn_sensor_init)(xmedia_u32 dev, xmedia_sensor_init_mode init_mode);
xmedia_s32 (*pfn_sensor_exit)(xmedia_u32 dev);
xmedia_s32 (*pfn_sensor_set_attr)(xmedia_u32 dev, const xmedia_sensor_attr *sns_attr);
xmedia_s32 (*pfn_sensor_get_property)(xmedia_u32 dev, xmedia_sensor_property *property);
xmedia_s32 (*pfn_sensor_set_work_mode)(xmedia_u32 dev, xmedia_sensor_work_mode
work_mode);
xmedia_s32 (*pfn_sensor_set_mipi_lanes)(xmedia_u32 dev, xmedia_sensor_mipi_lanes
mipi_lanes);

xmedia_s32 (*pfn_sensor_mirror)(xmedia_u32 dev, xmedia_bool mirror_en);
xmedia_s32 (*pfn_sensor_flip)(xmedia_u32 dev, xmedia_bool flip_en);

xmedia_s32 (*pfn_sensor_set_bus_info)(xmedia_u32 dev, const xmedia_sensor_commbus
*sns_bus_info);
xmedia_s32 (*pfn_sensor_set_dev_addr)(xmedia_u32 dev, xmedia_u32 slave_addr);
xmedia_s32 (*pfn_sensor_start)(xmedia_u32 dev);
xmedia_s32 (*pfn_sensor_stop)(xmedia_u32 dev);
xmedia_s32 (*pfn_sensor_standby)(xmedia_u32 dev);
xmedia_s32 (*pfn_sensor_resume)(xmedia_u32 dev);
xmedia_s32 (*pfn_sensor_write_reg)(xmedia_u32 dev, xmedia_u32 addr, xmedia_u32 data);
xmedia_s32 (*pfn_sensor_read_reg)(xmedia_u32 dev, xmedia_u32 addr, xmedia_u32 *data);
xmedia_s32 (*pfn_sensor_set_init_param)(xmedia_u32 dev, const xmedia_sensor_init_param
*def_cfg);
xmedia_s32 (*pfn_sensor_low_power)(xmedia_u32 dev, xmedia_bool enable);
} xmedia_sensor_register_isp_func;

```

【成员】

成员名称	描述
pfn_sensor_get_isp_default	获取 ISP 基础算法的初始值的回调函数指针。
pfn_sensor_get_isp_black_level	获取 sensor 的黑电平值的回调函数指针，支持根据 sensor 增益动态调整黑电平值。若此处动

	态调整黑电平值，则外部只能通过接口 xmedia_isp_set_black_level_attr 的手动模式设置黑电平值。
pfn_sensor_get_reg_info	获取 sensor 寄存器信息的回调函数指针，用于实现内核态配置 AE 信息。
pfn_sensor_get_bayer_pattern	获取 bayer 图像格式的回调函数指针。
pfn_sensor_get_capability	获取 sensor 的能力集的回调函数指针。
pfn_sensor_get_fps	获取 sensor 帧率(实时/最大/最小)的回调函数指针。
pfn_sensor_get_hmax_vmax	获取 sensor 的帧长行长的回调函数指针。
pfn_sensor_get_trig_attr	从模式 sensor 同步信号的回调函数指针。
pfn_sensor_get_dev_addr	获取 sensor 从设备地址。
pfn_sensor_update_reg_info	更新 update 标志的回调函数指针。 使用当前帧计算出的新的寄存器参数和上一帧的寄存器参数相比，值不同的寄存器对应的 update 标志将会被置为 true。
pfn_sensor_get_pdaf_info	获取 sensor pdaf 相关参数的回调函数指针。
pfn_sensor_format_pdaf	格式化 pdaf 统计信息的回调函数指针。
pfn_sensor_calc_image_size	根据 pd 宽高计算出对应的图像宽高的回调函数指针。
pfn_sensor_set_bus_info	配置 sensor 控制接口的回调函数指针。
pfn_sensor_init	初始化 sensor 的回调函数指针。
pfn_sensor_exit	sensor 的回调退出函数指针。
pfn_sensor_set_attr	设置 sensor 属性函数指针。
pfn_sensor_get_property	获取 sensor 当前的固有属性，包括支持的最大分辨率，帧率，输入时钟等属性。
pfn_sensor_set_work_mode	配置 sensor 工作模式：

	0: DEFAULT - sensor 默认工作模式, trigger 信号 sensor 自主提供 1: MASTER - 主从模式下配置, sensor 工作在 master 模式, 向 slave sensor 提供 trigger 信号 2: MASTER_SLAVE - 主从模式下配置, sensor 工作在 slave 模式, 接收 master sensor 提供 trigger 信号 4: HOST_SLAVE: 双从模式下配置, host(常见为 SoC)给 slave sensor 提供 trigger 信号
pfn_sensor_set_mipi_lanes	配置 mipi 通道数的回调函数指针。
pfn_sensor_mirror	设置 sensor 镜像模式的回调函数指针。
pfn_sensor_flip	设置 sensor 翻转模式的回调函数指针。
pfn_sensor_set_dev_addr	配置 sensor 寻址地址的回调函数指针。
pfn_sensor_start	启动 sensor 的回调函数指针。
pfn_sensor_stop	停止 sensor 的回调函数指针。
pfn_sensor_standby	设置 sensor 待机模式的回调函数指针。
pfn_sensor_resume	sensor 从待机模式恢复正常工作的回调函数指针。
pfn_sensor_write_reg	写 sensor 寄存器的回调函数指针。
pfn_sensor_read_reg	读 sensor 寄存器的回调函数指针。
pfn_sensor_set_init_param	配置 AE/AWB 默认相关参数的回调函数指针。
pfn_sensor_low_power	设置 sensor 低功耗模式的回调函数指针。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_sensor_register_awb_func

【说明】

定义 AWB 回调函数结构体信息。

【定义】

```
typedef struct {  
    xmedia_s32 (*pfn_sensor_get_awb_default)(xmedia_u32 dev,  
xmedia_isp_awb_sensor_default *awb_sns_dft);  
    xmedia_s32 (*pfn_sensor_set_color_temperature)(xmedia_u32 dev, xmedia_u32 ct);  
} xmedia_sensor_register_awb_func;
```

【成员】

成员名称	描述
pfn_sensor_get_awb_default	获取 AWB 默认配置的回调函数指针。
pfn_sensor_set_color_temperature	配置 AWB 色温。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_sensor_register_ae_func

【说明】

定义 AE 回调函数结构体信息。

【定义】

```
typedef struct {  
    xmedia_s32 (*pfn_sensor_get_ae_default)(xmedia_u32 dev, xmedia_isp_ae_sensor_default  
*ae_sns_dft);  
    xmedia_s32 (*pfn_sensor_set_fps)(xmedia_u32 dev, xmedia_float fps,  
xmedia_isp_ae_sensor_default *ae_sns_dft);  
    xmedia_s32 (*pfn_sensor_set_slow_framerate)(xmedia_u32 dev, xmedia_u32 full_lines,  
xmedia_isp_ae_sensor_default *ae_sns_dft);  
    xmedia_s32 (*pfn_sensor_update_inttime)(xmedia_u32 dev, xmedia_u32 int_time);  
    xmedia_s32 (*pfn_sensor_calc_again)(xmedia_u32 dev, xmedia_u32 *again_lin, xmedia_u32
```

```

*again_db);
    xmedia_s32 (*pfn_sensor_calc_dgain)(xmedia_u32 dev, xmedia_u32 *dgain_lin, xmedia_u32
*dgain_db);
    xmedia_s32 (*pfn_sensor_update_gains)(xmedia_u32 dev, xmedia_u32 again, xmedia_u32
dgain);
    xmedia_s32 (*pfn_sensor_get_max_inttime)(xmedia_u32 dev, xmedia_sensor_ae_inttime_attr
*inittime_attr);
    xmedia_s32 (*pfn_sensor_set_ae_wdr_attr)(xmedia_u32 dev, xmedia_isp_ae_wdr_mode
ae_wdr_mode);
} xmedia_sensor_register_ae_func;

```

【成员】

成员名称	描述
pfn_sensor_get_ae_default	获取 ae 默认参数的回调函数指针。
pfn_sensor_set_fps	设置 sensor 帧率的回调函数指针。
pfn_sensor_set_slow_framerate	设置 sensor 降帧帧率的回调函数指针。
pfn_sensor_update_inttime	更新 sensor 曝光时间的回调函数指针。
pfn_sensor_calc_again	计算 sensor 模拟增益的回调函数指针。
pfn_sensor_calc_dgain	计算 sensor 数字增益的回调函数指针。
pfn_sensor_update_gains	更新 sensor 增益的回调函数指针。
pfn_sensor_get_max_inttime	获取 sensor 最大曝光时间的回调函数指针。
pfn_sensor_set_ae_wdr_attr	设置 AE WDR 属性的回调函数指针。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_sensor_register_info

【说明】

定义 sensor 注册信息。

【定义】

```
typedef struct {
    xmedia_u32 dev_id;
    xmedia_char sensor_name[XMEDIA_SENSOR_NAME_MAX_SIZE];
    xmedia_sensor_register_isp_func isp_func;
    xmedia_sensor_register_awb_func awb_func;
    xmedia_sensor_register_ae_func ae_func;
} xmedia_sensor_register_info;
```



说明

```
#define XMEDIA_SENSOR_NAME_MAX_SIZE 128
```

【成员】

成员名称	描述
dev_id	sensor 的设备号。
sensor_name	sensor 的名称。
isp_func	sensor 提供给 isp 库的回调函数结构体。
awb_func	sensor 提供给 awb 库的回调函数结构体。
ae_func	sensor 提供给 ae 库的回调函数结构体。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_ae_register_info

【说明】

定义 AE 库向 ISP 的注册信息。

【定义】

```
typedef struct {
    xmedia_char alg_name[XMEDIA_ISP_ALG_NAME_MAX_SIZE];
    xmedia_u32 alg_id;
    xmedia_s32 (*pfn_ae_init)(xmedia_u32 channel, const xmedia_isp_ae_param *ae_param);
    xmedia_s32 (*pfn_ae_run)(xmedia_u32 channel, const xmedia_isp_ae_info *ae_info,
```

```

xmedia_isp_ae_result *ae_result);
    xmedia_s32 (*pfn_ae_ctrl)(xmedia_u32 channel, xmedia_u32 cmd, xmedia_void *value);
    xmedia_s32 (*pfn_ae_exit)(xmedia_u32 channel);
} xmedia_isp_ae_register_info;

```

【成员】

成员名称	描述
alg_name	ae 向 isp 注册的算法库的名称。
alg_id	ae 向 isp 注册的算法 id。
pfn_ae_init	ae 向 isp 提供的算法初始化回调函数指针。
pfn_ae_run	ae 向 isp 提供的算法运行回调函数指针。
pfn_ae_ctrl	ae 向 isp 提供的算法控制回调函数指针。
pfn_ae_exit	ae 向 isp 提供的算法退出回调函数指针。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_ae_info

【说明】

定义 ae 库 run 函数需要的统计信息。

【定义】

```

typedef struct {
    xmedia_u64                frame_cnt;
    xmedia_isp_smart_info     smart_info;
    xmedia_u16                fe_ae_row_num;
    xmedia_u16                fe_ae_col_num;
    xmedia_isp_fe_ae_stat_hist fe_ae_stat_hist;
    xmedia_isp_fe_ae_stat_global_avg *fe_ae_stat_global_avg;
    xmedia_isp_fe_ae_stat_zone_avg  *fe_ae_stat_zone_avg;
    xmedia_isp_fe_ae_blend_stat  *fe_ae_blend_stat;
}

```

```

xmedia_u16                be_ae_row_num;
xmedia_u16                be_ae_col_num;
xmedia_isp_be_ae_stat_hist be_ae_stat_hist;
xmedia_isp_be_ae_stat_global_avg *be_ae_stat_global_avg;
xmedia_isp_be_ae_stat_zone_avg  *be_ae_stat_zone_avg;
xmedia_isp_be_ae_blend_stat    *be_ae_blend_stat;
} xmedia_isp_ae_info;

```

【成员】

成员名称	描述
frame_cnt	帧数。
smart_info	感兴趣区域信息。
fe_ae_row_num	fe 分块行数。
fe_ae_col_num	fe 分块列数。
fe_ae_stat_hist	fe 直方图统计信息。
fe_ae_stat_global_avg	fe 全局均值统计信息。
fe_ae_stat_zone_avg	fe 分块均值统计信息。
fe_ae_blend_stat	fe 多通道融合统计信息。
be_ae_row_num	be 分块行数。
be_ae_col_num	be 分块列数。
be_ae_stat_hist	be 直方图统计信息。
be_ae_stat_global_avg	be 全局均值统计信息。
be_ae_stat_zone_avg	be 分块均值统计信息。
be_ae_blend_stat	be 多通道融合统计信息。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_ae_result

【说明】

定义 ae 库 run 函数输出的配置信息。

【定义】

```
typedef struct {
    xmedia_u32 int_time[XMEDIA_ISP_WDR_CHN_MAX_NUM];
    xmedia_u32 isp_dgain;
    xmedia_u32 again;
    xmedia_u32 dgain;
    xmedia_u32 iso;
    xmedia_u32 isp_dgain_short_frame;
    xmedia_u32 again_short_frame;
    xmedia_u32 dgain_short_frame;
    xmedia_u32 iso_short_frame;
    xmedia_u32 isp_dgain_sep_thd;
    xmedia_u8  ae_run_interval;

    xmedia_bool piris_valid;
    xmedia_s32  piris_pos;
    xmedia_u32  piris_gain;

    xmedia_isp_wdr_mode wdr_mode;
    xmedia_u32          max_sensor_again;
    xmedia_u32          wdr_gain[XMEDIA_ISP_WDR_CHN_MAX_NUM];
    xmedia_u32          hmax_times;
    xmedia_u32          vmax;

    xmedia_isp_ae_stat_attr stat_attr;
    // xmedia_isp_dcf_info    dcf_update_info;
} xmedia_isp_ae_result;
```

【成员】

成员名称	描述
int_time	sensor 的曝光时间，单位：us。
isp_dgain	图像补偿增益配置参数，256 为 1 倍。
again	sensor 的模拟增益，1024 为 1 倍。

成员名称	描述
dgain	sensor 的数字增益，1024 为 1 倍。
iso	iso 等于 sensor 模拟增益倍数 x sensor 数字增益倍数 x isp 数字增益倍数 x 100，iso 值 100 为 1 倍。
isp_dgain_short_frame	短帧图像补偿增益配置参数，256 为 1 倍。
again_short_frame	短帧 sensor 的模拟增益，1024 为 1 倍。
dgain_short_frame	短帧 sensor 的数字增益，1024 为 1 倍。
iso_short_frame	短帧 iso 等于 sensor 模拟增益倍数 x sensor 数字增益倍数 x isp 数字增益倍数 x 100，iso 值 100 为 1 倍。。
isp_dgain_sep_thd	ainr 场景下，进 npu 之前的 isp_dgn 倍数，剩余的分配给 npu 之后的 isp_dgn，256 为 1 倍。
ae_run_interval	ae 计算间隔帧数。
piris_valid	光圈开关。
piris_pos	光圈位置。
piris_gain	光圈增益。
wdr_mode	wdr 模式。
max_sensor_again	开启 fpn 校正的最大 sensor 模拟增益。
wdr_gain	wdr 模式下长短帧的补偿增益。
hmax_times	sensor 读出一行的时间，单位 us。
vmax	Sesnor 行数。
stat_attr	配置统计信息参数。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_sensor_register_ae_info

【说明】

定义 sensor 向 AE 库的注册信息。

【定义】

```
typedef struct {
    xmedia_u32          dev_id;
    xmedia_sensor_register_ae_func func;
} xmedia_sensor_register_ae_info;
```

【成员】

成员名称	描述
dev_id	sensor 设备号。
func	sensor 提供给 ae 库的回调函数结构体。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_awb_register_info

【说明】

定义 AWB 库向 ISP 库的注册信息。

【定义】

```
typedef struct {
    xmedia_char alg_name[XMEDIA_ISP_ALG_NAME_MAX_SIZE];
    xmedia_u32  alg_id;
    xmedia_s32 (*pfn_awb_init)(xmedia_u32 channel, const xmedia_isp_awb_param
*awb_param);
    xmedia_s32 (*pfn_awb_run)(xmedia_u32 channel, const xmedia_isp_awb_info *awb_info,
xmedia_isp_awb_result *awb_result);
    xmedia_s32 (*pfn_awb_ctrl)(xmedia_u32 channel, xmedia_u32 cmd, xmedia_void *value);
    xmedia_s32 (*pfn_awb_exit)(xmedia_u32 channel);
} xmedia_isp_awb_register_info;
```

【成员】

成员名称	描述
alg_name	awb 向 isp 注册的算法库的名称。
alg_id	awb 向 isp 注册的算法 id。
pfn_awb_init	awb 向 isp 提供的算法初始化回调函数指针。
pfn_awb_run	awb 向 isp 提供的算法运行回调函数指针。
pfn_awb_ctrl	awb 向 isp 提供的算法控制回调函数指针。
pfn_awb_exit	awb 向 isp 提供的算法退出回调函数指针。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_awb_info

【说明】

定义 isp 提供给 AWB 的统计信息结构体。

【定义】

```
typedef struct {  
    xmedia_u32                frame_cnt;  
    xmedia_isp_awb_stat_global awb_stat_global;  
    xmedia_isp_awb_stat_local_avg_r *awb_stat_local_avg_r;  
    xmedia_isp_awb_stat_local_avg_g *awb_stat_local_avg_g;  
    xmedia_isp_awb_stat_local_avg_b *awb_stat_local_avg_b;  
    xmedia_isp_awb_stat_local_valid_count *stat_local_valid_count;  
    xmedia_isp_awb_blend_stat *awb_blend_stat_local;  
    xmedia_u8                awb_gain_switch;  
    xmedia_u8                awb_stat_switch;  
    xmedia_u32                wdr_wb_gain[XMEDIA_ISP_BAYER_PATTERN_NUM];  
    xmedia_u8                awb_zone_row;  
    xmedia_u8                awb_zone_col;  
    xmedia_u16               awb_width;
```

```

xmedia_u16          awb_height;
} xmedia_isp_awb_info;

```

【成员】

成员名称	描述
frame_cnt	帧计数,取值范围[0,0xFFFFFFFF]
awb_stat_global	awb 全局均值统计,取值范围[0,65535]
awb_stat_local_avg_r	awb 分块均值 R 通道,取值范围[0,65535]
awb_stat_local_avg_g	awb 分块均值 G 通道,取值范围[0,65535]
awb_stat_local_avg_b	awb 分块均值 B 通道,取值范围[0,65535]
stat_local_valid_count	awb 分块均值个数统计, 取值范围[0,0xFFFFFFFF]
awb_blend_stat_local	blend 模式下的分块均值统计,取值范围[0,65535]
awb_gain_switch	awb,取值范围[0,65535] 增益模式切换, 取值[0,1] 0: 白平衡增益配置在 WDR 合成前的 DG1 1: 白平衡配置在 wb 处; 只支持配置在 wb 处;
awb_stat_switch	awb 统计位置切换, 取值范围[0,2] 0: 白平衡统计模块在 EXPANDER 后; 1: 白平衡统计模块在 DGN2 后; 2: 白平衡统计模块在 DRC 后。
wdr_wb_gain	在 WDR 合成前的 DG1 处配置的白平衡增益值。 暂未支持;
awb_zone_row	awb 行分块个数,取值范围[1,32]
awb_zone_col	awb 列分块个数,取值范围[1,32]
awb_width	分辨率宽。
awb_height	分辨率高。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_esp_awb_result

【说明】

定义 awb 库输出结果。

【定义】

```
typedef struct {  
    xmedia_u32  
    white_balance_gain[XMEDIA_ESP_BAYER_PATTERN_NUM];  
    xmedia_u16          ccm[XMEDIA_ESP_CCM_MATRIX_SIZE];  
    xmedia_u32          ct;  
    xmedia_u8           saturation;  
    xmedia_bool         ccm_en;  
    xmedia_esp_awb_raw_stat_attr raw_stat_attr;  
} xmedia_esp_awb_result;
```

【成员】

成员名称	描述
white_balance_gain	wb gain (R G G B),取值范围[0,4095]。
ccm	ccm 矩阵结果,取值范围[0,4095]。
ct	色温值,取值范围[0,65535]。
saturation	饱和度,取值范围[0,255]。
ccm_en	ccm 使能开关,取值范围[0,1]。
raw_stat_attr	raw 统计相关配置。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_esp_awb_raw_stat_attr

【说明】

awb 库输出结果，raw_attr 修改统计相关配置。

【定义】

```
typedef struct {  
    xmedia_bool change;  
    xmedia_bool black_white_update;  
    xmedia_u16  metering_white_level_awb;  
    xmedia_u16  metering_black_level_awb;  
    xmedia_bool cbcr_update;  
    xmedia_u16  metering_crref_max_awb;  
    xmedia_u16  metering_cbref_max_awb;  
    xmedia_u16  metering_crref_min_awb;  
    xmedia_u16  metering_cbref_min_awb;  
    xmedia_bool pos_update;  
    xmedia_esp_awb_switch pos_switch;  
} xmedia_esp_awb_raw_stat_attr;
```

【成员】

成员名称	描述
change	是否改变标志位,取值范围[0,1]。
black_white_update	亮暗值是否改变标志位,取值范围[0,1]。
metering_white_level_awb	最亮值阈值配置,取值范围[0,65535]。
metering_black_level_awb	最暗值阈值配置,取值范围[0,65535]。
cbcr_update	cbcr 是否改变标志位,取值范围[0,1]。
metering_crref_max_awb	cr max 配置(r/g),取值范围[0,4095]。
metering_cbref_max_awb	cb max 配置(b/g),取值范围[0,4095]。
metering_crref_min_awb	cr min 配置(r/g),取值范围[0,4095]。
metering_cbref_min_awb	cb min 配置(b/g),取值范围[0,4095]。
pos_update	位置是否改变标志位,取值范围[0,1]。

成员名称	描述
pos_switch	统计位置切换,取值范围[0,2]。 0: 白平衡统计模块在 EXPANDER 后; 1: 白平衡统计模块在 DGN2 后; 2: 白平衡统计模块在 DRC 后。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_sensor_register_awb_info

【说明】

定义 sensor 向 AWB 库的注册信息结构体。

【定义】

```
typedef struct {
    xmedia_u32 dev_id;
    xmedia_sensor_register_awb_func func;
} xmedia_sensor_register_awb_info;
```

【成员】

成员名称	描述
dev_id	sensor 的设备号。
func	sensor 提供给 awb 库的回调函数结构体。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_sensor_register_af_info

【说明】

定义 sensor 向 AF 库的注册信息。

【定义】

```
typedef struct {
    xmedia_u32 dev_id;
    xmedia_sensor_register_af_func func;
} xmedia_sensor_register_af_info;
```

【成员】

成员名称	描述
dev_id	sensor 的设备号。
func	预留的 sensor 提供给 AF 库的回调函数结构体。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_af_register_info

【说明】

定义 AF 库向 ISP 库的注册信息。

【定义】

```
typedef struct {
    xmedia_char alg_name[XMEDIA_ISP_ALG_NAME_MAX_SIZE];
    xmedia_u32 alg_id;
    xmedia_s32 (*pfn_af_init)(xmedia_u32 channel, const xmedia_isp_af_param *af_param,
xmedia_isp_af_result *af_result);
    xmedia_s32 (*pfn_af_run)(xmedia_u32 channel, const xmedia_isp_af_info *af_info,
xmedia_isp_af_result *af_result);
    xmedia_s32 (*pfn_af_ctrl)(xmedia_u32 channel, xmedia_u32 cmd, xmedia_void *value);
```

```
xmedia_s32 (*pfn_af_exit)(xmedia_u32 channel);
} xmedia_esp_af_register_info;
```

【成员】

成员名称	描述
alg_name	af 向 isp 注册的算法库的名称。
alg_id	af 向 isp 注册的算法 id。
pfn_af_init	af 向 isp 提供的算法初始化回调函数指针。
pfn_af_run	af 向 isp 提供的算法运行回调函数指针。
pfn_af_ctrl	af 向 isp 提供的算法控制回调函数指针。
pfn_af_exit	af 向 isp 提供的算法退出回调函数指针。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_esp_af_param

【说明】

定义 ISP 提供给 AF 库的初始化参数。

【定义】

```
typedef struct {
    xmedia_bool    pdaf_en;
    xmedia_bool    cdaf_en;
    xmedia_video_size image_size;
} xmedia_esp_af_param;
```

【成员】

成员名称	描述
pdaf_en	pdaf 使能开关。

成员名称	描述
cdaf_en	cdaf 使能开关。
image_size	图像的尺寸大小。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_af_result

【说明】

定义 AF 库返给 ISP 的参数。

【定义】

```
typedef struct {
    xmedia_bool    pdaf_en;
    xmedia_bool    cdaf_en;
    xmedia_bool    roi_update;
    xmedia_video_rect roi;
} xmedia_isp_af_result;
```

【成员】

成员名称	描述
pdaf_en	pdaf 使能开关。
cdaf_en	cdaf 使能开关。
roi_update	感兴趣区域更新标志。
roi	感兴趣区域。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_af_info

【说明】

定义 ISP 提供给 AF 库的统计信息相关参数。

【定义】

```
typedef struct {
    xmedia_u64                frame_cnt;
    xmedia_video_rect         roi;
    xmedia_isp_af_pdaf_statistics pdaf_stat;
    const xmedia_isp_fe_af_zone_matrix *fe_zone_matrix;
    const xmedia_isp_be_af_zone_matrix *be_zone_matrix;
} xmedia_isp_af_info;
```

【成员】

成员名称	描述
frame_cnt	帧计数。
roi	实际生效的感兴趣区域。
pdaf_stat	pdaf 统计信息。
fe_zone_matrix	cdaf fe 分块统计信息。
be_zone_matrix	cdaf be 分块统计信息。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_af_pdaf_statistics

【说明】

定义 ISP 提供给 AF 库的 pdaf 统计信息相关参数。

【定义】

```
typedef struct {  
    xmedia_u64    stat_frame_cnt;  
    xmedia_u32    width;  
    xmedia_u32    height;  
    xmedia_u32    bit_width;  
    xmedia_void *pd_stat;  
} xmedia_isp_af_pdaf_statistics;
```

【成员】

成员名称	描述
stat_frame_cnt	统计信息对应的帧计数。
width	pd 统计信息的总宽度。
height	pd 统计信息的总高度。
bit_width	pd 统计信息的位宽。
pd_stat	pd 统计信息。

【注意事项】

pd 统计信息在内存中按类型依次排列，同一种类型的统计信息集中排列。

比如 pd 统计信息只有左点和右点两种类型，前半段内存保存左点，后半段内存保存右点。此时左右点的宽为 width，高为 height/2。

【相关数据类型及接口】

无

xmedia_isp_fe_af_zone_matrix

【说明】

定义 ISP 提供给 AF 库的 cdaf fe 统计信息相关参数。

【定义】

```
typedef struct {  
    xmedia_isp_af_zone  
    zone_matrix[XMEDIA_ISP_WDR_CHN_MAX_NUM][XMEDIA_ISP_AF_ZONE_ROW_MAX_NUM]  
                                                [XMEDIA_ISP_AF_ZONE_COL_MAX_NUM];  
}
```

```
} xmedia_isp_fe_af_zone_matrix;
```

【成员】

成员名称	描述
zone_matrix	cdaf 分块统计信息。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_be_af_zone_matrix

【说明】

定义 ISP 提供给 AF 库的 cdaf fe 统计信息相关参数。

【定义】

```
typedef struct {  
    xmedia_isp_af_zone  
    zone_matrix[XMEDIA_ISP_AF_ZONE_ROW_MAX_NUM][XMEDIA_ISP_AF_ZONE_COL_MAX_N  
    UM];  
} xmedia_isp_be_af_zone_matrix;
```

【成员】

成员名称	描述
zone_matrix	cdaf 分块统计信息。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_lens_register_info

【说明】

定义 LENS 向 ISP 注册的信息。

【定义】

```
typedef struct {
    xmedia_u32 dev_id;
    xmedia_char lens_name[XMEDIA_LENS_NAME_MAX_SIZE];
    xmedia_lens_register_isp_func isp_func;
    xmedia_lens_register_awb_func awb_func;
    xmedia_lens_register_ae_func ae_func;
    xmedia_lens_register_af_func af_func;
} xmedia_lens_register_info;
```

【成员】

成员名称	描述
dev_id	lens 的设备号。
lens_name	lens 的名字。
isp_func	lens 提供给 isp 的接口。
awb_func	lens 提供给 awb 库的接口。
ae_func	lens 提供给 ae 库的接口。
af_func	lens 提供给 af 库的接口。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_lens_register_isp_func

【说明】

定义 LENS 向 ISP 注册的回调函数。

【定义】

```
typedef struct {  
    xmedia_s32 (*pfn_lens_init)(xmedia_u32 dev);  
    xmedia_s32 (*pfn_lens_exit)(xmedia_u32 dev);  
    xmedia_s32 (*pfn_lens_standby)(xmedia_u32 dev);  
    xmedia_s32 (*pfn_lens_resume)(xmedia_u32 dev);  
    xmedia_s32 (*pfn_lens_park)(xmedia_u32 dev);  
    xmedia_s32 (*pfn_lens_set_ircut)(xmedia_u32 dev, xmedia_bool enable);  
} xmedia_lens_register_isp_func;
```

【成员】

成员名称	描述
pfn_lens_init	lens 初始化的回调函数指针。
pfn_lens_exit	lens 反初始化的回调函数指针。
pfn_lens_standby	lens 待机的回调函数指针。
pfn_lens_resume	lens 恢复运行的回调函数指针。
pfn_lens_park	lens 归位的回调函数指针。
pfn_lens_set_ircut	设置 lens ircut 开关状态的回调函数指针。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_lens_register_ae_info

【说明】

定义 LENS 向 AE 库的注册信息。

【定义】

```
typedef struct {  
    xmedia_u32 dev_id;  
    xmedia_lens_register_ae_func func;
```

```
} xmedia_lens_register_ae_info;
```

【成员】

成员名称	描述
dev_id	lens 的设备号。
func	lens 提供给 AE 库的回调函数结构体。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_lens_register_ae_func

【说明】

定义 LENS 向 AE 库注册的回调函数。

【定义】

```
typedef struct {  
    xmedia_s32 (*pfn_lens_set_aperture)(xmedia_u32 dev, xmedia_u32 index);  
    xmedia_s32 (*pfn_lens_set_shutter)(xmedia_u32 dev, xmedia_bool enable);  
} xmedia_lens_register_ae_func;
```

【成员】

成员名称	描述
pfn_lens_set_aperture	设置镜头光圈的回调函数指针。
pfn_lens_set_shutter	设置镜头快门的回调函数指针。

【注意事项】

无

【相关数据类型及接口】

[xmedia_lens_register_ae_info](#)

xmedia_lens_register_awb_info

【说明】

定义 LENS 向 AWB 库的注册信息。

【定义】

```
typedef struct {  
    xmedia_u32          dev_id;  
    xmedia_lens_register_awb_func func;  
} xmedia_lens_register_awb_info;
```

【成员】

成员名称	描述
dev_id	lens 的设备号。
func	预留的 lens 提供给 AWB 库的回调函数结构体。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_lens_register_af_info

【说明】

定义 LENS 向 AF 库的注册信息。

【定义】

```
typedef struct {  
    xmedia_u32          dev_id;  
    xmedia\_lens\_register\_af\_func func;  
} xmedia_lens_register_af_info;
```

【成员】

成员名称	描述
dev_id	lens 的设备号。
func	lens 提供给 AF 库的回调函数结构体。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_lens_register_af_func

【说明】

定义 lens 向 AF 库注册的回调函数结构体。

【定义】

```
typedef struct {
    xmedia_s32 (*pfn_lens_zoom_in)(xmedia_u32 dev, xmedia_u32 distance);
    xmedia_s32 (*pfn_lens_zoom_out)(xmedia_u32 dev, xmedia_u32 distance);
    xmedia_s32 (*pfn_lens_zoom_pi)(xmedia_u32 dev, xmedia_u32 pi);
    xmedia_s32 (*pfn_lens_zoom_status)(xmedia_u32 dev);
    xmedia_s32 (*pfn_lens_focus_near)(xmedia_u32 dev, xmedia_u32 distance);
    xmedia_s32 (*pfn_lens_focus_far)(xmedia_u32 dev, xmedia_u32 distance);
    xmedia_s32 (*pfn_lens_focus_pi)(xmedia_u32 dev, xmedia_u32 pi);
    xmedia_s32 (*pfn_lens_focus_status)(xmedia_u32 dev);
} xmedia_lens_register_af_func;
```

【成员】

成员名称	描述
pfn_lens_zoom_in	变倍放大的回调函数指针。
pfn_lens_zoom_out	变倍缩小的回调函数指针。
pfn_lens_zoom_pi	变倍镜头回到 pi 点的回调函数指针。
pfn_lens_zoom_status	获取变倍镜头的工作状态的回调函数指针。
pfn_lens_focus_near	对焦镜头向 near 端移动的回调函数指针。

成员名称	描述
pfn_lens_focus_far	对焦镜头向 far 端移动的回调函数指针。
pfn_lens_focus_pi	对焦镜头回到 pi 点的回调函数指针。
pfn_lens_focus_status	获取对焦镜头的工作状态的回调函数指针。

【注意事项】

如果回调函数指针不需要赋值，需要置为 NULL。

【相关数据类型及接口】

[xmedia_lens_register_af_info](#)

xmedia_isp_diff_attr

【说明】

定义多路 ISP 差异属性信息。

【定义】

```
typedef struct {
    xmedia_s32 blc_offset[XMEDIA_ISP_BAYER_PATTERN_NUM];
    xmedia_u16 ct_num;
    xmedia_u16 ct_level[XMEDIA_ISP_CT_MAX_COUNT];
    xmedia_s32
wb_slope[XMEDIA_ISP_CT_MAX_COUNT][XMEDIA_ISP_BAYER_PATTERN_NUM];
    xmedia_s32
wb_offset[XMEDIA_ISP_CT_MAX_COUNT][XMEDIA_ISP_BAYER_PATTERN_NUM];
    xmedia_u16 ccm[XMEDIA_ISP_CT_MAX_COUNT][XMEDIA_ISP_CCM_MATRIX_SIZE];
    xmedia_s16 luma_differ;
} xmedia_isp_diff_attr;
```



说明

```
#define XMEDIA_ISP_CCM_MATRIX_SIZE    9
#define XMEDIA_ISP_CT_MAX_COUNT      16
```

【成员】

成员名称	描述
blc_offset	多路 blc 差异化配置，由差异化标定给出， 取值范围[-4095,4095]

成员名称	描述
ct_num	有效色温数量，由差异化标定给出，取值范围[0, 15]
ct_level	色温数组，由差异化标定给出，取值范围[0, 65535]
wb_slope	多路 gain slope 差异化配置，由差异化标定给出，取值范围[-1048575,1048575]
wb_offset	多路 gain offset 差异化配置，由差异化标定给出，取值范围[-1048575,1048575]
ccm	多路 ccm 差异化配置，由差异化标定给出，取值范围[0,65535]
luma_differ	多路亮度差异化配置，取值范围[-128,128]

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_wait_type

【说明】

定义 ISP 中断类型。

【定义】

```
typedef enum {
    XMEDIA_ISP_WAIT_TYPE_FE_START = 0,
    XMEDIA_ISP_WAIT_TYPE_FE_END,
    XMEDIA_ISP_WAIT_TYPE_BE_END,
    XMEDIA_ISP_WAIT_TYPE_MAX
} xmedia_isp_wait_type;
```

【成员】

成员名称	描述
XMEDIA_ISP_WAIT_TYPE_FE_START	FE 帧起始中断。
XMEDIA_ISP_WAIT_TYPE_FE_END	FE 帧结束中断。
XMEDIA_ISP_WAIT_TYPE_BE_END	BE 帧结束中断。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_ctrl_param

【说明】

定义 ISP 控制信息。

【定义】

```
typedef struct {
    xmedia_u32 proc_interval;
    xmedia_u32 stat_interval;
    xmedia_u32 int_timeout;
} xmedia_isp_ctrl_param;
```

【成员】

成员名称	描述
proc_interval	Proc 信息更新频率。
stat_interval	统计信息更新频率。
int_timeout	超时时间。

【注意事项】

int_timeout 参数暂未支持。

【相关数据类型及接口】

无

xmedia_isp_iris_f_no

【说明】

定义 ISP 光圈 F 值。

【定义】

```
typedef enum {  
    XMEDIA_ISP_IRIS_F_NO_32_0 = 0,  
    XMEDIA_ISP_IRIS_F_NO_22_0,  
    XMEDIA_ISP_IRIS_F_NO_16_0,  
    XMEDIA_ISP_IRIS_F_NO_11_0,  
    XMEDIA_ISP_IRIS_F_NO_8_0,  
    XMEDIA_ISP_IRIS_F_NO_5_6,  
    XMEDIA_ISP_IRIS_F_NO_4_0,  
    XMEDIA_ISP_IRIS_F_NO_2_8,  
    XMEDIA_ISP_IRIS_F_NO_2_0,  
    XMEDIA_ISP_IRIS_F_NO_1_4,  
    XMEDIA_ISP_IRIS_F_NO_1_0,  
    XMEDIA_ISP_IRIS_F_NO_MAX  
} xmedia_isp_iris_f_no;
```

【成员】

成员名称	描述	等效增益
XMEDIA_ISP_IRIS_F_NO_32_0	光圈 F32.0。	1
XMEDIA_ISP_IRIS_F_NO_22_0	光圈 F22.0。	2
XMEDIA_ISP_IRIS_F_NO_16_0,	光圈 F16.0。	4
XMEDIA_ISP_IRIS_F_NO_11_0,	光圈 F11.0。	8
XMEDIA_ISP_IRIS_F_NO_8_0,	光圈 F8.0。	16
XMEDIA_ISP_IRIS_F_NO_5_6,	光圈 F5.6。	32
XMEDIA_ISP_IRIS_F_NO_4_0,	光圈 F4.0。	64
XMEDIA_ISP_IRIS_F_NO_2_8,	光圈 F2.8。	128
XMEDIA_ISP_IRIS_F_NO_2_0,	光圈 F2.0。	256
XMEDIA_ISP_IRIS_F_NO_1_4,	光圈 F1.4。	512

成员名称	描述	等效增益
XMEDIA_ISP_IRIS_F_NO_1_0,	光圈 F1.0。	1024

【注意事项】

无。

【相关数据类型及接口】

无。

xmedia_isp_smart_roi

【说明】

定义 ISP 提供给 AE 的人形、人脸统计信息结构体

【定义】

```
typedef struct {
    xmedia_bool enable;
    xmedia_bool available;
    xmedia_u8    luma;
} xmedia_isp_smart_roi;
```

【成员】

成员名称	描述
enable	模型是否使能。
available	模型是否有可用的检测结果。
luma	模型检测结果的整体亮度。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_smart_info

【说明】

定义 ISP 智能信息。

【定义】

```
typedef struct {  
    xmedia_isp_smart_roi roi[XMEDIA_ISP_SMART_CLASS_MAX];  
} xmedia_isp_smart_info;
```

 说明
宏定义如下：

```
#define XMEDIA_ISP_SMART_CLASS_MAX    2
```

【成员】

成员名称	描述
roi	用户感兴趣区域。

【注意事项】

无

【相关数据类型及接口】

无

3 SENSOR

3.1 功能概述

SENSOR 为图像采集硬件设备，在自身能力集范围内，可采集用户目标分辨率、帧率的图像数据供 ISP 处理，同时 SENSOR 驱动对 ISP 和 3A 提供相应的控制接口，用于实现自动曝光，自动白平衡，自动聚焦相关功能。

3.2 API 参考

- [xmedia_sensor_get_property](#): 获取当前规格下 sensor 的属性。
- [xmedia_sensor_set_work_mode](#): 设置 sensor 工作模式。
- [xmedia_sensor_set_attr](#): 配置 sensor 的工作属性。
- [xmedia_sensor_set_mipi_lanes](#): 配置 MIPI Lane 数。
- [xmedia_sensor_set_dev_addr](#): 配置 sensor 的寻址地址。
- [xmedia_sensor_set_trig_signal](#): SLAVE 从模式下，配置 sensor 与同步触发信号的绑定关系。
- [xmedia_sensor_init](#): 初始化 sensor。
- [xmedia_sensor_exit](#): 反初始化 sensor。
- [xmedia_sensor_start](#): 启动 sensor。
- [xmedia_sensor_stop](#): 停止 sensor。
- [xmedia_sensor_standby](#): 待机 sensor。
- [xmedia_sensor_resume](#): 对 standby 状态下的 sensor 进行恢复操作。
- [xmedia_sensor_mirror](#): 配置 sensor 镜像模式。
- [xmedia_sensor_flip](#): 配置 sensor 翻转模式。
- [xmedia_sensor_write_reg](#): 写 sensor 寄存器。
- [xmedia_sensor_read_reg](#): 读 sensor 寄存器。
- [xmedia_sensor_set_init_param](#): 配置 AE/AWB 相关默认参数。

- `xmedia_sensor_get_safety_status`: 获取 sensor 的车规安全状态及信息。
- `xmedia_sensor_create_group`: 创建 sensor 功能组。
- `xmedia_sensor_destroy_group`: 销毁 sensor 功能组。

`xmedia_sensor_get_property`

【描述】

获取当前规格下 sensor 的属性。

【语法】

```
xmedia_s32 xmedia_sensor_get_property(xmedia_u32 pipe, xmedia_sensor_property *property);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
property	sensor 当前规格下的属性配置。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: `xmedia_isp.h`
- 库文件: `libxmedia_isp.a`

【注意】

无

`xmedia_sensor_set_work_mode`

【描述】

配置 sensor 当前工作模式。

【语法】

```
xmedia_s32 xmedia_sensor_set_work_mode(xmedia_u32 pipe, xmedia_sensor_work_mode mode);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
mode	sensor 的工作模式。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

xmedia_sensor_set_attr

【描述】

配置 sensor 图像属性。

【语法】

```
xmedia_s32 xmedia_sensor_set_attr(xmedia_u32 pipe, const xmedia_sensor_attr *sns_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
sns_attr	Sensor 图像属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

xmedia_sensor_set_mipi_lanes

【描述】

MIPI 协议传输图像数据时，配置 MIPI Lane 数。

【语法】

```
xmedia_s32 xmedia_sensor_set_mipi_lanes(xmedia_u32 pipe, xmedia_sensor_mipi_lanes mipi_lanes);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
mipi_lanes	sensor 的 mipi 数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

xmedia_sensor_set_dev_addr

【描述】

配置 sensor 的寻址地址。

【语法】

```
xmedia_s32 xmedia_sensor_set_dev_addr(xmedia_u32 pipe, xmedia_u32 slave_addr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
slave_addr	sensor 寻址地址。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

xmedia_sensor_set_trig_signal

【描述】

SLAVE 模式下, 配置 Sensor 与触发信号的绑定关系。

【语法】

```
xmedia_s32 xmedia_sensor_set_trig_signal(xmedia_u32 pipe, xmedia_sensor_trig_signal *signal);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
signal	Sensor 触发信号属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败, 其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

xmedia_sensor_init

【描述】

初始化 sensor 配置。

【语法】

```
xmedia_s32 xmedia_sensor_init(xmedia_u32 pipe, xmedia_sensor_config *cfg);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
cfg	sensor 初始化配置参数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

xmedia_sensor_exit

【描述】

反初始化 sensor。

【语法】

```
xmedia_s32 xmedia_sensor_exit(xmedia_u32 pipe);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_isp.h
- 库文件：libxmedia_isp.a

【注意】

无

xmedia_sensor_start

【描述】

启动 sensor。

【语法】

```
xmedia_s32 xmedia_sensor_start(xmedia_u32 pipe);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

xmedia_sensor_stop

【描述】

停止 sensor。

【语法】

```
xmedia_s32 xmedia_sensor_stop(xmedia_u32 pipe);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h

- 库文件: libxmedia_isp.a

【注意】

无

xmedia_sensor_standby

【描述】

待机 sensor。

【语法】

```
xmedia_s32 xmedia_sensor_standby(xmedia_u32 pipe);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败, 其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

xmedia_sensor_resume

【描述】

对 standby 状态下的 sensor 进行恢复操作。

【语法】

```
xmedia_s32 xmedia_sensor_resume(xmedia_u32 pipe);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

xmedia_sensor_mirror

【描述】

配置 sensor 镜像模式。

【语法】

```
xmedia_s32 xmedia_sensor_mirror(xmedia_u32 pipe, xmedia_bool enable);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
enable	镜像模式使能开关。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

xmedia_sensor_flip

【描述】

配置 sensor 翻转模式。

【语法】

```
xmedia_s32 xmedia_sensor_flip(xmedia_u32 pipe, xmedia_bool enable);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
enable	翻转模式使能开关。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

xmedia_sensor_write_reg

【描述】

写 sensor 寄存器。

【语法】

```
xmedia_s32 xmedia_sensor_write_reg(xmedia_u32 pipe, xmedia_u32 addr, xmedia_u32 data);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
addr	待写寄存器的地址。	输入
data	待写寄存器的值。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败, 其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

xmedia_sensor_read_reg

【描述】

读 sensor 寄存器。

【语法】

```
xmedia_s32 xmedia_sensor_read_reg(xmedia_u32 pipe, xmedia_u32 addr, xmedia_u32 *data);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
addr	待读 sensor 寄存器的地址。	输入
data	待读 sensor 寄存器的值。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

xmedia_sensor_set_init_param

【描述】

配置 AE/AWB 相关默认参数。

【语法】

```
xmedia_s32 xmedia_sensor_set_init_param(xmedia_u32 pipe, const xmedia_sensor_init_param *init_param);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
init_param	AE/AWB 初始化参数结构体。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

xmedia_sensor_get_safety_status

【描述】

获取 sensor 车规安全状态及信息

【语法】

```
xmedia_s32 xmedia_sensor_get_safety_status(xmedia_u32 pipe, xmedia_u32 *status, xmedia_u32 *info);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
status	当前的安全状态，其值可能是一种错误状态或几种错误状态的位或。	输出
info	与异常状态相关的详细信息	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

暂未支持。

xmedia_sensor_create_group

【描述】

创建 sensor 功能组。对功能组内任一 sensor 的操作，都将引起对组内其他 sensor 的联动操作，这些操作包括：启动、停止、待机、恢复和改变帧率。通过 sensor 功能组以及功能组内 sensor 的联动操作，可以实现功能组内多个 sensor 的从模式同步控制功能。

【语法】

```
xmedia_s32 xmedia_sensor_create_group(xmedia_sensor_group_config *group_cfg,
xmedia_handle *group);
```

【参数】

参数名称	描述	输入/输出
group_cfg	group 功能组的参数配置。	输入
group	group 功能组的 handle。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

该接口非必调接口，用于多路输入时，同一类型功能的 sensor 创建功能组，以便于同步和同一操作；[xmedia_sensor_init](#) 后 [xmedia_sensor_start](#) 前调用，如果是 HOST_SLAVE/MASTER_SLAVE 的功能组，group 的 pipe_num 数量不能低于 [XMEDIA_SENSOR_GROUP_MIN_PIPE_NUM](#)；[xmedia_sensor_init](#) 后 [xmedia_sensor_start](#) 前调用，如果是 HOST_SLAVE/MASTER_SLAVE 的功能组，需要先通过 [xmedia_sensor_set_work_mode](#)、[xmedia_sensor_set_trig_signal](#) 配置主从模式和触发信号。

xmedia_sensor_destroy_group

【描述】

销毁 sensor 功能组。在销毁功能组后，对任一 sensor 的操作，不再会进行启动，停止等操作上的联动。

【语法】

```
xmedia_s32 xmedia_sensor_destroy_group(xmedia_handle group);
```

【参数】

参数名称	描述	输入/输出
group	group 功能组的 handle	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

该接口非必调接口，如果创建了 group 功能组，需要功能组内每个 sensor 在 [xmedia_sensor_stop](#) 后 [xmedia_sensor_exit](#) 前调用本接口。

3.3 数据类型

- [xmedia_video_wdr_format](#): 定义 WDR 模式下数据传输格式。
- [xmedia_sensor_clock_freq](#): 定义 SENSOR 工作时钟频率。
- [xmedia_intf_type](#): 定义 SENSOR 数据的接口类型。
- [xmedia_video_data_width](#): 定义 SENSOR 输出数据位宽。
- [xmedia_sensor_property](#): 定义 SENSOR 属性。
- [xmedia_sensor_work_mode](#): 定义 SENSOR 的工作模式。
- [xmedia_sensor_attr](#): 定义 sensor 图像属性。
- [xmedia_sensor_mipi_lanes](#): 定义 sensor 传输 mipi lane 数。
- [xmedia_sensor_init_mode](#): 定义 sensor 初始化模式。
- [xmedia_sensor_ctrl_sig](#): 定义 sensor 控制信号。
- [xmedia_sensor_bus_type](#): 定义 sensor 通讯类型。

- `xmedia_sensor_commbus`: 定义 sensor 通讯接口方式。
- `xmedia_sensor_config`: 定义 sensor 配置属性。
- `xmedia_sensor_init_param`: 定义 sensor 初始化配置属性。
- `xmedia_sensor_trig_attr`: 定义 sensor slave 模式同步信号触发属性。
- `xmedia_sensor_trig_source`: 定义 slave sensor 触发信号源。
- `xmedia_sensor_trig_signal`: 定义 sensor 触发信号属性。
- `xmedia_sensor_group_type`: 定义 sensor 功能组类型。
- `xmedia_sensor_group_config`: 定义 sensor 功能组属性配置。
- `xmedia_sensor_pdaf_info`: 定义 sensor pdaf 相关信息。
- `xmedia_sensor_pdaf_feature`: 定义 sensor pdaf 特性。
- `xmedia_sensor_pdaf_type`: 定义 sensor pdaf 类型。
- `xmedia_sensor_pdaf_correction`: 定义 sensor pdaf 矫正位置。
- `xmedia_sensor_pdaf_pattern`: 定义 sensor pdaf 模式。
- `xmedia_sensor_pdaf_pattern_type`: 定义 sensor pdaf 模式类型。
- `xmedia_sensor_pdaf_point`: 定义 sensor pdaf 点的坐标信息。
- `xmedia_isp_point`: 定义 ISP 点的坐标信息。

`xmedia_video_wdr_format`

【说明】

定义 WDR 模式下数据传输方式。

【定义】

```
typedef enum {
    XMEDIA_VIDEO_WDR_FMT_NONE = 0x0,
    XMEDIA_VIDEO_WDR_FMT_VC,
    XMEDIA_VIDEO_WDR_FMT_DOL,
    XMEDIA_VIDEO_WDR_FMT_BUILTIN,
    XMEDIA_VIDEO_WDR_FMT_MAX,
} xmedia_video_wdr_format;
```

【成员】

成员名称	描述
<code>XMEDIA_VIDEO_WDR_FMT_NONE</code>	线性模式的数据格式。

成员名称	描述
XMEDIA_VIDEO_WDR_FMT_VC	以 vc 号来区分长短帧数据（长短帧有独立的 FS/FE 数据包）。
XMEDIA_VIDEO_WDR_FMT_DOL	以特殊标识码来区分长短帧数据（长短帧公用一组 FS/FE 数据包）
XMEDIA_VIDEO_WDR_FMT_BUILTIN	在 sensor 内部融合的 WDR 数据格式。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_get_property](#)

[xmedia_sensor_clock_freq](#)

【说明】

定义 sensor 工作时钟频率。

【定义】

```
typedef enum {
    XMEDIA_SENSOR_CLOCK_FREQ_74_25MHZ = 74250000,
    XMEDIA_SENSOR_CLOCK_FREQ_72MHZ    = 72000000,
    XMEDIA_SENSOR_CLOCK_FREQ_54MHZ    = 54000000,
    XMEDIA_SENSOR_CLOCK_FREQ_37_125MHZ = 37125000,
    XMEDIA_SENSOR_CLOCK_FREQ_36MHZ     = 36000000,
    XMEDIA_SENSOR_CLOCK_FREQ_27MHZ     = 27000000,
    XMEDIA_SENSOR_CLOCK_FREQ_24MHZ     = 24000000,
    XMEDIA_SENSOR_CLOCK_FREQ_12MHZ     = 12000000,
} xmedia_sensor_clock_freq;
```

【成员】

成员名称	描述
XMEDIA_SENSOR_CLOCK_FREQ_74_25MHZ	时钟为 74.25MHZ。
XMEDIA_SENSOR_CLOCK_FREQ_72MHZ	时钟为 72MHZ。
XMEDIA_SENSOR_CLOCK_FREQ_54MHZ	时钟为 54MHZ。

成员名称	描述
XMEDIA_SENSOR_CLOCK_FREQ_37_125MHZ	时钟为 37.125MHZ。
XMEDIA_SENSOR_CLOCK_FREQ_36MHZ	时钟为 36MHZ。
XMEDIA_SENSOR_CLOCK_FREQ_27MHZ	时钟为 27MHZ。
XMEDIA_SENSOR_CLOCK_FREQ_24MHZ	时钟为 24MHZ。
XMEDIA_SENSOR_CLOCK_FREQ_12MHZ	时钟为 12MHZ。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_get_property](#)

xmedia_intf_type

【说明】

定义 SENSOR 数据的传输协议类型。

【定义】

```
typedef enum {
    XMEDIA_INTF_TYPE_BT601 = 0,
    XMEDIA_INTF_TYPE_BT656,
    XMEDIA_INTF_TYPE_BT1120,
    XMEDIA_INTF_TYPE_MIPI_CSI,
    XMEDIA_INTF_TYPE_MIPI_DSI,
    XMEDIA_INTF_TYPE_LCD,
    XMEDIA_INTF_TYPE_LVDS,
    XMEDIA_INTF_TYPE_DC,
    XMEDIA_INTF_TYPE_MAX
} xmedia_intf_type;
```

【成员】

成员名称	描述
XMEDIA_INTF_TYPE_BT601	BT601 类型。
XMEDIA_INTF_TYPE_BT656	BT656 类型。

成员名称	描述
XMEDIA_INTF_TYPE_BT1120	BT1120 类型。
XMEDIA_INTF_TYPE_MIPI_CSI	MIPI_CSI 类型。
XMEDIA_INTF_TYPE_MIPI_DSI	MIPI_DSI 类型。
XMEDIA_INTF_TYPE_LCD	LCD 类型。
XMEDIA_INTF_TYPE_LVDS	LVDS 类型。
XMEDIA_INTF_TYPE_DC	DC 类型。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_get_property](#)

xmedia_video_data_width

【说明】

定义 SENSOR 输出数据位宽。

【定义】

```
typedef enum {
    XMEDIA_VIDEO_DATA_WIDTH_8 = 0x0,
    XMEDIA_VIDEO_DATA_WIDTH_10,
    XMEDIA_VIDEO_DATA_WIDTH_12,
    XMEDIA_VIDEO_DATA_WIDTH_14,
    XMEDIA_VIDEO_DATA_WIDTH_16,
    XMEDIA_VIDEO_DATA_WIDTH_MAX
} xmedia_video_data_width;
```

【成员】

成员名称	描述
XMEDIA_VIDEO_DATA_WIDTH_8	8bit 数据位宽。
XMEDIA_VIDEO_DATA_WIDTH_10	10bit 数据位宽。
XMEDIA_VIDEO_DATA_WIDTH_12	12bit 数据位宽。

成员名称	描述
XMEDIA_VIDEO_DATA_WIDTH_14	14bit 数据位宽。
XMEDIA_VIDEO_DATA_WIDTH_16	16bit 数据位宽。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_get_property](#)

xmedia_sensor_property

【说明】

定义 sensor 属性。

【定义】

```
typedef struct {
    xmedia_u32          width;
    xmedia_u32          height;
    xmedia\_video\_wdr\_mode wdr_mode;

    xmedia\_video\_wdr\_format wdr_format;
    xmedia_float        max_fps;
    xmedia\_sensor\_clock\_freq input_clock;

    xmedia\_intf\_type        output_intf;
    xmedia\_video\_pixel\_format pixel_format;
    xmedia\_video\_data\_width bit_width;

    xmedia\_video\_bayer\_format bayer_format[XMEDIA_SENSOR_MIRROR_FLIP_TYPE_MAX];

    xmedia_u8   vcid[XMEDIA_SENSOR_VCID_MAX_NUM];
    xmedia_u32 bit_rate; // unit: Mbps
} xmedia_sensor_property;
```



说明

宏定义如下：

```
#define XMEDIA_SENSOR_MIRROR_FLIP_TYPE_MAX    4
#define XMEDIA_SENSOR_VCID_MAX_NUM           4
```

【成员】

成员名称	描述
width	sensor 宽。
height	sensor 高。
wdr_mode	Sensor wdr 模式。
wdr_format	Sensor wdr 数据传输格式。
max_fps	sensor 最大帧率。
input_clock	sensor 输入时钟频率。
output_intf	sensor 数据的传输接口类型。
pixel_format	sensor 数据的像素格式。
bit_width	sensor 数据的位宽。
bayer_format	sensor 数据的 bayer 格式。
vcid	sensor 的 vc id 号。
bit_rate	MIPI 协议传输时，每条 lanes 上的传输速率。 单位 Mbps。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_get_property](#)

xmedia_sensor_work_mode

【说明】

定义 SENSOR 的工作模式。

【定义】

```
typedef enum {
    XMEDIA_SENSOR_WORK_MODE_NORMAL = 0,
    XMEDIA_SENSOR_WORK_MODE_MASTER,
    XMEDIA_SENSOR_WORK_MODE_SLAVE,
    XMEDIA_SENSOR_WORK_MODE_MAX,
```

```
} xmedia_sensor_work_mode;
```

【成员】

成员名称	描述
XMEDIA_SENSOR_WORK_MODE_NORMAL	sensor 常规工作模式，也是默认工作模式。
XMEDIA_SENSOR_WORK_MODE_MASTER	sensor 工作在 master 模式，向外提供 VSYNC 等同步信号。
XMEDIA_SENSOR_WORK_MODE_SLAVE	sensor 工作在 slave 模式，受外部 VSYNC 信号触发曝光和数据输出。
XMEDIA_SENSOR_WORK_MODE_MAX	Sensor 工作模式无效值。

【注意事项】

Sensor 工作在从模式，且需芯片提供触发信号时，需调用 [xmedia_sensor_set_trig_signal](#) 接口触发信号源相关的配置。

【相关数据类型及接口】

[xmedia_sensor_set_work_mode](#)

[xmedia_sensor_set_trig_signal](#)

xmedia_sensor_attr

【说明】

定义 sensor 图像属性。

【定义】

```
typedef struct {  
    xmedia_u16 width;  
    xmedia_u16 height;  
    xmedia_u8 wdr_mode;  
} xmedia_sensor_attr;
```

【成员】

成员名称	描述
width	sensor 输出图像的宽。

成员名称	描述
height	sensor 输出图像的高。
wdr_mode	sensor 输出图像的 WDR 模式。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_set_attr](#)

xmedia_sensor_mipi_lanes

【说明】

定义 sensor 输出图像的 mipi lane 数。

【定义】

```
typedef enum {
    XMEDIA_SENSOR_MIPI_LANES_1L = 1,
    XMEDIA_SENSOR_MIPI_LANES_2L = 2,
    XMEDIA_SENSOR_MIPI_LANES_4L = 4,
    XMEDIA_SENSOR_MIPI_LANES_8L = 8,
    XMEDIA_SENSOR_MIPI_LANES_MAX
} xmedia_sensor_mipi_lanes;
```

【成员】

成员名称	描述
XMEDIA_SENSOR_MIPI_LANES_1L	1 lanes。
XMEDIA_SENSOR_MIPI_LANES_2L	2 lanes。
XMEDIA_SENSOR_MIPI_LANES_4L	4 lanes。
XMEDIA_SENSOR_MIPI_LANES_8L	8 lanes。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_set_mipi_lanes](#)

xmedia_sensor_init_mode

【说明】

定义 sensor 初始化模式。

【定义】

```
typedef enum {  
    XMEDIA_SENSOR_INIT_MODE_NORMAL = 0,  
    XMEDIA_SENSOR_INIT_MODE_SKIP,  
    XMEDIA_SENSOR_INIT_MODE_MAX,  
} xmedia_sensor_init_mode;
```

【成员】

成员名称	描述
XMEDIA_SENSOR_INIT_MODE_NORMAL	正常进行 sensor 初始化。
XMEDIA_SENSOR_INIT_MODE_SKIP	用于快启，配置为此模式时，可跳过 sensor 序列初始化。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_init](#)

xmedia_sensor_ctrl_sig

【说明】

定义 sensor 控制信号。

【定义】

```
typedef struct {  
    xmedia_u32 clk_ch;  
    xmedia_u32 rst_ch;  
} xmedia_sensor_ctrl_sig;
```

【成员】

成员名称	描述
clk_ch	sensor mclk 时钟通道。
rst_ch	sensor rst 通道。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_init](#)

xmedia_sensor_bus_type

【说明】

定义 sensor 通讯类型。

【定义】

```
typedef enum {
    XMEDIA_SENSOR_BUS_TYPE_I2C = 0,
    XMEDIA_SENSOR_BUS_TYPE_SPI,
    XMEDIA_SENSOR_BUS_TYPE_MAX,
} xmedia_sensor_bus_type;
```

【成员】

成员名称	描述
XMEDIA_SENSOR_BUS_TYPE_I2C	I2C 通讯。
XMEDIA_SENSOR_BUS_TYPE_SPI	SPI 通讯。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_init](#)

xmedia_sensor_commbus

【说明】

定义 sensor 通讯接口方式。

【定义】

```
typedef struct {  
    xmedia_sensor_bus_type type;  
  
    union {  
        xmedia_s8 i2c_dev;  
  
        struct {  
            xmedia_u8 spi_dev;  
            xmedia_u8 spi_cs; // chip select:  
        } spi_info;  
    };  
} xmedia_sensor_commbus;
```

【成员】

成员名称	描述
type	sensor 通讯类型。
i2c_dev	i2c 通讯的设备号。
spi_info.spi_dev	spi 通讯的设备号。
spi_info.spi_cs	spi 通讯的片选信号。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_init](#)

[xmedia_sensor_config](#)

【说明】

定义 sensor 配置属性。

【定义】

```
typedef struct {  
    xmedia_sensor_init_mode init_mode;  
    xmedia_sensor_ctrl_sig ctrl_sig;  
    xmedia_sensor_commbus comm_bus;  
} xmedia_sensor_config;
```

【成员】

成员名称	描述
init_mode	sensor 初始化模式。
ctrl_sig	sensor 控制信号。
comm_bus	sensor 通讯总线。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_init](#)

xmedia_sensor_init_param

【说明】

定义 sensor 初始化配置属性。

【定义】

```
typedef struct {  
    xmedia_bool ir_mode;  
    xmedia_u32  exp_time;  
    xmedia_u32  again;  
    xmedia_u32  dgain;  
    xmedia_u32  ispdgain;  
    xmedia_u32  exposure;  
    xmedia_u32  init_iso;  
    xmedia_u32  piris_fno;  
    xmedia_u16  wb_rgain;  
    xmedia_u16  wb_ggain;  
    xmedia_u16  wb_bgain;  
    xmedia_u16  sample_rgain;  
    xmedia_u16  sample_bgain;  
    xmedia_u16  ccm[XMEDIA_SENSOR_CCM_MATRIX_SIZE];  
} xmedia_sensor_init_param;
```

📖 说明

宏定义如下：

```
#define XMEDIA_SENSOR_CCM_MATRIX_SIZE 9
```

【成员】

成员名称	描述
ir_mode	红外补光灯模式开关。
exp_time	曝光时间。
again	模拟增益。
dgain	数字增益。
ispdgain	ISP 增益。
exposure	曝光量。
init_iso	初始化 ISO。
piris_fno	光圈大小。
wb_rgain	AWB 的 R 通道增益。
wb_ggain	AWB 的 G 通道增益。
wb_bgain	AWB 的 B 通道增益。
sample_rgain	当前样机 AWB 在线标定得到的 G/R 的值。
sample_bgain	当前样机 AWB 在线标定得到的 G/B 的值。
ccm	CCM 矩阵。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_set_init_param](#)

[xmedia_isp_get_sensor_init_param](#)

[xmedia_sensor_trig_attr](#)

【说明】

定义 sensor slave 模式同步信号触发属性。

【定义】

```
typedef struct {
    xmedia_u16 trig_dev;
```

```

xmedia_u16 trig_mode;
xmedia_u8  hs_enable;
xmedia_u8  vs_enable;
xmedia_u8  hs_invert;
xmedia_u8  vs_invert;
xmedia_u32 hs_time;
xmedia_u32 vs_time;
xmedia_u32 hs_active;
xmedia_u32 vs_active;
xmedia_u32 hs_delay;
xmedia_u32 ph_offset;
xmedia_u32 delay_frame_num;
} xmedia_sensor_trig_attr;

```

【成员】

成员名称	描述
trig_dev	触发信号设备配置通道。
trig_mode	工作模式，0：同步模式；1：相位模式
hs_enable	XHS 输出使能
vs_enable	XVS 输出使能
hs_invert	XHS 极性配置，1：低电平有效；0：高电平有效
vs_invert	XVS 极性配置，1：低电平有效；0：高电平有效
hs_time	XHS 信号周期，单位：us。
vs_time	XVS 信号周期，单位：us。
hs_active	XHS 有效电平宽度，单位：us。
vs_active	XVS 有效电平宽度，单位：us。
hs_delay	XHS 相对 XVS 信号的延迟时间，单位：us。
ph_offset	相位偏移。
delay_frame_num	触发信号延迟生效帧数。

【注意事项】

sensor 从模式下，需芯片提供触发信号时，需实现 sensor 驱动的回调函数 pfn_sensor_get_trig_attr。

【相关数据类型及接口】

[xmedia_sensor_set_trig_signal](#)

xmedia_sensor_trig_source

【说明】

定义 slave sensor 触发信号源。

【定义】

```
typedef enum{
    XMEDIA_SENSOR_TRIG_SOURCE_NONE = 0,
    XMEDIA_SENSOR_TRIG_SOURCE_MASTER,
    XMEDIA_SENSOR_TRIG_SOURCE_HOST,
    XMEDIA_SENSOR_TRIG_SOURCE_MAX,
}xmedia_sensor_trig_source;
```

【成员】

成员名称	描述
XMEDIA_SENSOR_TRIG_SOURCE_NONE	无触发信号源。
XMEDIA_SENSOR_TRIG_SOURCE_MASTER	触发信号源为 master sensor。
XMEDIA_SENSOR_TRIG_SOURCE_HOST	触发信号源为 host。
XMEDIA_SENSOR_TRIG_SOURCE_MAX	触发信号源无效值。

【注意事项】

无。

【相关数据类型及接口】

[xmedia_sensor_set_trig_signal](#)

xmedia_sensor_trig_signal

【说明】

定义 sensor 触发信号属性。

【定义】

```
typedef struct{
    xmedia_sensor_trig_source source;
    xmedia_u32 dev_id;
}xmedia_sensor_trig_signal;
```

【成员】

成员名称	描述
source	触发信号源。
dev_id	trig 设备号，salve sensor 触发信号源为 host 时，需配置此参数。

【注意事项】

无。

【相关数据类型及接口】

[xmedia_sensor_set_trig_signal](#)

xmedia_sensor_group_type

【说明】

定义 SENSOR 功能组类型。

【定义】

```
typedef enum{
    XMEDIA_SENSOR_GROUP_TYPE_MASTER_SLAVE,
    XMEDIA_SENSOR_GROUP_TYPE_HOST_SLAVE,
    XMEDIA_SENSOR_GROUP_TYPE_MAX,
} xmedia_sensor_group_type;
```

【成员】

成员名称	描述
XMEDIA_SENSOR_GROUP_TYPE_MASTER_SLAVE	由 master sensor 和 slave sensor 组成的功能组。
XMEDIA_SENSOR_GROUP_TYPE_HOST_SLAVE	由 host 和 slave sensor 组成的功能组。

成员名称	描述
XMEDIA_SENSOR_GROUP_TYPE_MAX	SENSOR 功能组类型无效值。

【注意事项】

无。

【相关数据类型及接口】

[xmedia_sensor_create_group](#)

xmedia_sensor_group_config

【说明】

定义 sensor 功能组配置属性。

【定义】

```
typedef struct{
    xmedia_sensor_group_type type;
    xmedia_u32 pipe_num;
    xmedia_u32 pipe[XMEDIA_SENSOR_GROUP_MAX_PIPE_NUM];
}xmedia_sensor_group_config;
```

说明

宏定义如下：

```
#define XMEDIA_SENSOR_GROUP_MAX_PIPE_NUM 4
#define XMEDIA_SENSOR_GROUP_MIN_PIPE_NUM 2
```

【成员】

成员名称	描述
type	Sensor 功能组类型。
pipe_num	Sensor 功能组内 pipe 数量。
pipe	Sensor 功能组内 pipe 号。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_create_group](#)

xmedia_sensor_pdaf_info

【说明】

定义 sensor pdaf 相关信息。

【定义】

```
typedef struct {
    const xmedia_sensor_pdaf_feature *pdaf_feature;
    const xmedia_sensor_pdaf_point *pdaf_point;
} xmedia_sensor_pdaf_info;
```

【成员】

成员名称	描述
pdaf_feature	Sensor pdaf 特性。
pdaf_point	Sensor pdaf 点坐标。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_sensor_pdaf_feature

【说明】

定义 sensor pdaf 特性。

【定义】

```
typedef struct {
    xmedia_sensor_pdaf_type pd_type;
    xmedia_sensor_pdaf_correction pd_correction;
    xmedia_u8 mipi_vcid;
    xmedia_u8 mipi_data_type;
    xmedia_u8 pd_data_width;
    xmedia_video_rect pd_whole_pixel_area;
    xmedia_video_size pd_block_pixel_size;
    xmedia_video_size pd_block_data_size;
    xmedia_sensor_pdaf_pattern pd_block_pattern;
} xmedia_sensor_pdaf_feature;
```

【成员】

成员名称	描述
pd_type	pd 类型。
pd_correction	pd 矫正位置。
mipi_vcid	传输 pd 数据的虚拟通道号。
mipi_data_type	传输 pd 数据的 mipi 数据类型。
pd_data_width	pd 数据的位宽。
pd_whole_pixel_area	包含 pd 数据的整个像素区域。
pd_block_pixel_size	单个 pd 块的像素宽高。
pd_block_data_size	单个 pd 块里的 pd 数据的宽高。
pd_block_pattern	单个 pd 块里 pd 的排布模式。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_pdaf_info](#)

xmedia_sensor_pdaf_type

【说明】

定义 sensor pdaf 类型。

【定义】

```
typedef enum {  
    XMEDIA_SENSOR_PDAF_TYPE_NONE,  
    XMEDIA_SENSOR_PDAF_TYPE_1,  
    XMEDIA_SENSOR_PDAF_TYPE_2,  
    XMEDIA_SENSOR_PDAF_TYPE_3,  
    XMEDIA_SENSOR_PDAF_TYPE_MAX,  
} xmedia_sensor_pdaf_type;
```

【成员】

成员名称	描述
XMEDIA_SENSOR_PD_AF_TYPE_1	SENSOR 通过 MIPI VC/DT 输出分块区域的 PHASE DIFFERENCE 和 CONFIDENCE LEVEL。
XMEDIA_SENSOR_PD_AF_TYPE_2	SENSOR 通过 MIPI VC/DT 输出图像区域内的 PD DATA。
XMEDIA_SENSOR_PD_AF_TYPE_3	PD DATA 嵌入在图像数据中一同输出。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_pdaf_feature](#)

[xmedia_sensor_pdaf_correction](#)

【说明】

定义 sensor pdaf 矫正位置。

【定义】

```
typedef enum {
    XMEDIA_SENSOR_PDAF_CORRECT_NONE,
    XMEDIA_SENSOR_PDAF_CORRECT_BY_SENSOR,
    XMEDIA_SENSOR_PDAF_CORRECT_BY_ISP,
    XMEDIA_SENSOR_PDAF_CORRECT_MAX,
} xmedia_sensor_pdaf_correction;
```

【成员】

成员名称	描述
XMEDIA_SENSOR_PD_AF_CORRECT_BY_SENSOR	由 sensor 矫正图像。
XMEDIA_SENSOR_PD_AF_CORRECT_BY_ISP	由 ISP 矫正图像。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_pdaf_feature](#)

xmedia_sensor_pdaf_pattern

【说明】

定义 sensor pdaf 模式。

【定义】

```
typedef struct {
    xmedia_sensor_pdaf_pattern_type type;
    xmedia_u32 cycle;
    xmedia_u32
    pattern[XMEDIA_SENSOR_MIRROR_FLIP_TYPE_MAX];
} xmedia_sensor_pdaf_pattern;
```

【成员】

成员名称	描述
type	pd 类型。
cycle	pd 点循环周期。
pattern	pd 点循环模式。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_pdaf_feature](#)

xmedia_sensor_pdaf_pattern_type

【说明】

定义 sensor pdaf 模式类型。

【定义】

```
typedef enum {
    XMEDIA_SENSOR_PDAF_PATTERN_UNDEFINED,
    XMEDIA_SENSOR_PDAF_PATTERN_LINE,
    XMEDIA_SENSOR_PDAF_PATTERN_PIXEL,
```

```

XMEDIA_SENSOR_PDAF_PATTERN_MAX,
} xmedia_sensor_pdaf_pattern_type;

```

【成员】

成员名称	描述
XMEDIA_SENSOR_PDAF_PATTERN_UNDEFINED	未定义的类型。
XMEDIA_SENSOR_PDAF_PATTERN_LINE	以 pd 行为单位进行循环。
XMEDIA_SENSOR_PDAF_PATTERN_PIXEL	以 pd 点为单位进行循环。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_pdaf_pattern](#)

[xmedia_sensor_pdaf_point](#)

【说明】

定义 sensor pdaf 点的坐标信息。

【定义】

```

typedef struct {
    xmedia_u32    lrpd_num;
    xmedia\_isp\_point left_pd_point[XMEDIA_SENSOR_PDAF_LRPD_MAX_NUM];
    xmedia\_isp\_point right_pd_point[XMEDIA_SENSOR_PDAF_LRPD_MAX_NUM];
} xmedia_sensor_pdaf_point;

```



说明

宏定义如下：

```
#define XMEDIA_SENSOR_PDAF_LRPD_MAX_NUM 64
```

【成员】

成员名称	描述
lrpd_num	pd 对的数量。
left_pd_point	左 pd 点的坐标。

成员名称	描述
right_pd_point	右 pd 点的坐标。

【注意事项】

无

【相关数据类型及接口】

[xmedia_sensor_pdaf_info](#)

xmedia_isp_point

【说明】

定义 ISP 点的坐标信息。

【定义】

```
typedef struct {
    xmedia_u16 x;
    xmedia_u16 y;
} xmedia_isp_point;
```

【成员】

成员名称	描述
x	横坐标。
y	纵坐标。

【注意事项】

无

【相关数据类型及接口】

4 LENS

4.1 功能概述

LENS 是一个能够接收光信号并汇聚光信号于感光器件的装置。LENS 驱动向 ISP 和 3A 提供了相应的控制接口，用以实现变倍、对焦和控制光圈的功能。

4.2 API 参考

- [xmedia_lens_init](#): 初始化 lens。
- [xmedia_lens_exit](#): 反初始化 lens。
- [xmedia_lens_standby](#): 待机 lens。
- [xmedia_lens_resume](#): 对 standby 状态下的 lens 进行恢复操作。

xmedia_lens_init

【描述】

初始化 lens。

【语法】

```
xmedia_s32 xmedia_lens_init(xmedia_u32 pipe);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a liblens.a

【注意】

无

xmedia_lens_exit

【描述】

反初始化 lens。

【语法】

```
xmedia_s32 xmedia_lens_exit(xmedia_u32 pipe);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a liblens.a

【注意】

无

xmedia_lens_standby

【描述】

待机 lens。

【语法】

```
xmedia_s32 xmedia_lens_standby(xmedia_u32 pipe);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a liblens.a

【注意】

无

xmedia_lens_resume

【描述】

对 standby 状态下的 lens 进行恢复操作。

【语法】

```
xmedia_s32 xmedia_lens_resume(xmedia_u32 pipe);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a liblens.a

【注意】

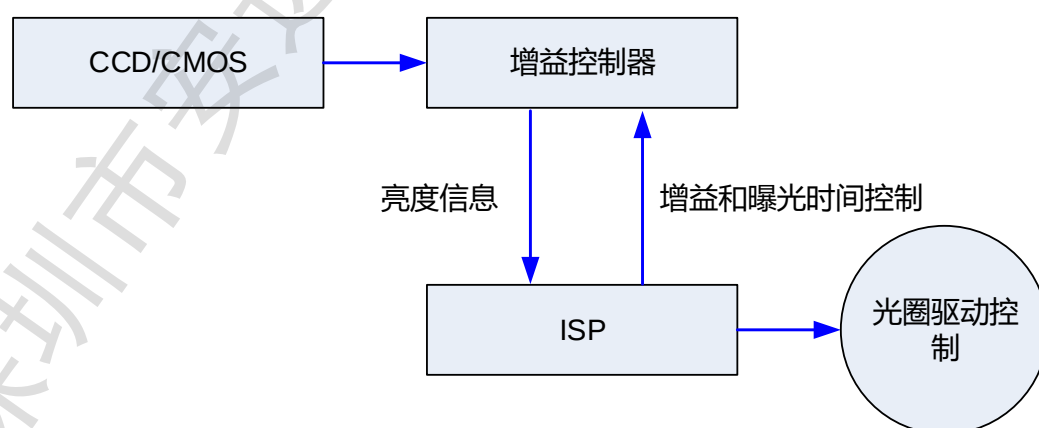
无

5 AE

5.1 概述

ISP AE 模块实现的功能是：根据自动测光系统获得当前图像的曝光量，再自动配置镜头光圈、sensor 快门及增益来获得最佳的图像质量。自动曝光的算法主要分光圈优先、快门优先、增益优先。光圈优先时算法会优先调整光圈到合适的位置，再分配曝光时间和增益，只适合 p-iris 镜头，这样能均衡噪声和景深。快门优先时算法会优先分配曝光时间，再分配 sensor 增益和 ISP 增益，这样拍摄的图像噪声会比较小。增益优先则是优先分配 sensor 增益和 ISP 增益，再分配曝光时间，适合拍摄运动物体的场景。当前 AE 算法也支持客户设定更灵活的曝光分配策略，AE 模块的工作流程如图 5-1 所示。

图5-1 AE 模块工作流程图



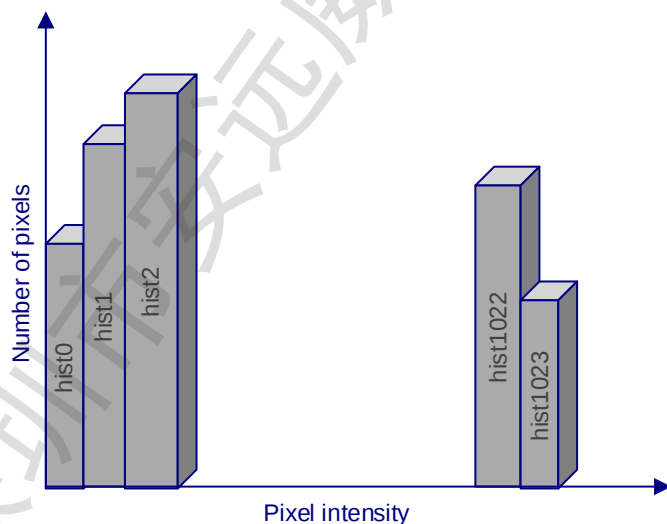
5.2 重要概念

- 曝光时间：sensor 积累电荷的时间，是 sensor pixel 从开始曝光到电量被读出的这段时间。
- 曝光增益：对 sensor 的输出电荷的总的放大系数，一般有数字增益和模拟增益，模拟增益引入的噪声会稍小，所以一般优先用模拟增益。
- 光圈：光圈是镜头中可以改变中间孔大小的机械装置。
- 抗闪烁：由于电灯电源工频与 sensor 的帧率不匹配而导致的画面闪烁，一般通过限定曝光时间和修改 sensor 的帧率来达到抗闪烁的效果。

5.3 功能描述

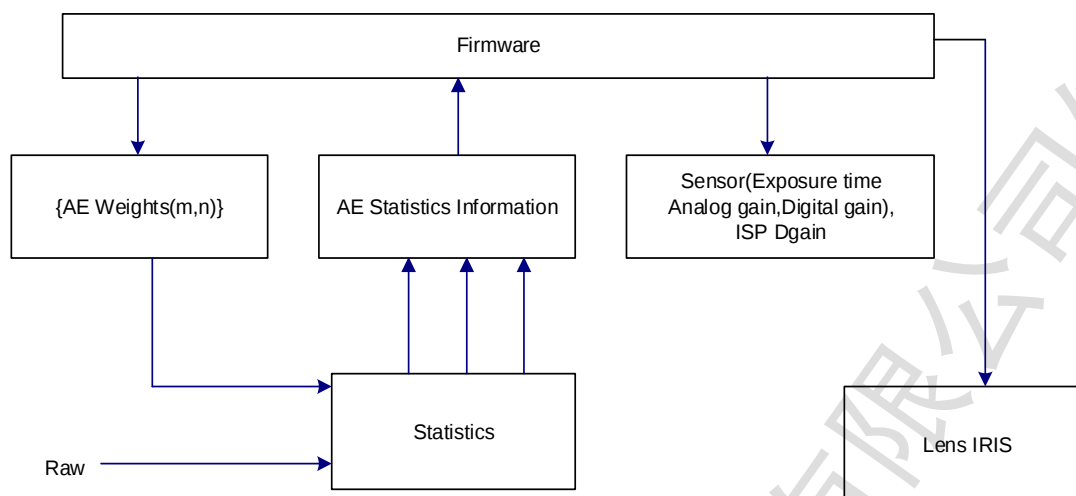
AE 模块主要有 ISP 的 AE 统计信息模块及 AE 控制策略的 AE 算法 Firmware 两部分组成。ISP 的 AE 统计信息模块主要是提供 sensor 输入数据的亮度信息统计。其提供的统计信息包括直方图和平均值，可同时提供整幅图像的 1024 段的直方图和 R/Gr/B/Gb 四分量平均值统计信息，还可提供将整幅图像分成 MxN 区块的每个区块的 R/Gr/B/Gb 四分量平均值统计信息，具体如图 5-2 所示。

图5-2 AE 1024 段统计信息直方图



AE 算法的主要工作原理是实时获取输入图像的统计信息并与设定目标亮度进行比较，而动态调节 sensor 的曝光时间和增益以及镜头光圈大小以达到实际亮度与设定目标亮度接近。其工作原理如图 5-3 所示。

图5-3 AE 工作原理图



5.4 API 参考

5.4.1 AE 库接口

所有 AE 库接口都只是针对 XMEDIA_AE 库，如果客户自己实现 AE 库，不需要关注这些接口，且无法使用这些接口。

该芯片 AE 库提供的注册接口包含两个功能：

其一，向 ISP 注册 AE 库；其二，ISP 向 AE 注册 sensor 接口；

该芯片 AE 库提供的反注册接口包含两个功能：

其一，向 ISP 反注册 AE 库；其二，实现 sensor 接口在 AE 库的反注册；

- [xmedia_ae_register](#)：向 ISP 注册 AE 库。
- [xmedia_ae_unregister](#)：向 ISP 反注册 AE 库。

xmedia_ae_register

【描述】

向 ISP 注册 AE 库。

【语法】

```
xmedia_s32 xmedia_ae_register(xmedia_u32 pipe);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_ae.h
- 库文件: libxmedia_isp.a、libxmedia_ae.a

【注意】

- 该接口调用了 ISP 库提供的 AE 注册回调接口 xmedia_isp_register_ae_lib，以实现 XMEDIA_AE 库向 ISP 库注册的功能。
- 每路 PIPE 支持注册一个 AE 库。
- 此接口不支持多进程操作。

【相关主题】

无

xmedia_ae_unregister

【描述】

向 ISP 反注册 AE 库。

【语法】

```
xmedia_s32 xmedia_ae_unregister(xmedia_u32 pipe);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_ae.h
- 库文件: libxmedia_isp.a、libxmedia_ae.a

【注意】

- 该接口调用了 ISP 库提供的 AE 反注册回调接口 xmedia_isp_unregister_ae_lib，以实现 AE 向 ISP 库反注册的功能。
- 此接口不支持多进程操作。

【相关主题】

无

5.4.2 AE 控制模块

曝光控制接口：

- [xmedia_ae_set_exposure_attr](#)：设置 AE 曝光属性。
- [xmedia_ae_get_exposure_attr](#)：获取 AE 曝光属性。
- [xmedia_ae_set_wdr_exposure_attr](#)：设置 WDR 模式下的 AE 曝光属性。
- [xmedia_ae_get_wdr_exposure_attr](#)：获取 WDR 模式下的 AE 曝光属性。
- [xmedia_ae_set_route_attr](#)：设置 AE 曝光分配策略。
- [xmedia_ae_get_route_attr](#)：获取 AE 曝光分配策略。
- [xmedia_ae_set_route_sf_attr](#)：设置短帧 AE 曝光分配策略。
- [xmedia_ae_get_route_sf_attr](#)：获取短帧 AE 曝光分配策略。

- `xmedia_ae_query_exposure_info`: 获取 ae 内部运行状态信息。
- `xmedia_ae_set_ext_route_attr`: 设置 AE 曝光分配策略扩展属性，支持分别设置 AE 分配策略中的 sensor 模拟增益，sensor 数字增益和 ISP 数字增益。
- `xmedia_ae_get_ext_route_attr`: 获取 AE 曝光分配策略扩展属性。
- `xmedia_ae_set_ext_route_sf_attr`: 设置短帧 AE 曝光分配策略扩展属性，支持分别设置 AE 分配策略中的 sensor 模拟增益，sensor 数字增益和 ISP 数字增益。。
- `xmedia_ae_get_ext_route_sf_attr`: 获取短帧 AE 曝光分配策略扩展属性。
- `xmedia_ae_set_smart_exposure_attr`: 设置 AE 智能曝光参数。
- `xmedia_ae_get_smart_exposure_attr`: 获取 AE 智能曝光参数。

`xmedia_ae_set_exposure_attr`

【描述】

设定 AE 曝光属性。

【语法】

```
xmedia_s32 xmedia_ae_set_exposure_attr(xmedia_u32 pipe, const xmedia_ae_exposure_attr
*exposure_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
exposure_attr	AE 曝光属性结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: `xmedia_ae.h`

•库文件: libxmedia_isp.a、libxmedia_ae.a

【注意】

- AE 曝光控制类型为自动时，曝光时间，曝光增益都由 AE 算法自动控制，可以通过配置自动曝光属性结构体 exposure_attr 里面的参数得到不同的曝光效果。
- AE 曝光控制类型为手动时，可以通过配置手动曝光属性结构体控制使能类型（曝光时间使能、sensor 模拟增益使能、sensor 数字增益使能、ISP 数字增益使能）及相应的曝光参数（曝光时间、sensor 模拟增益、sensor 数字增益、ISP 数字增益）。
- AE 曝光控制类型为自动时，配置手动曝光属性的参数无效。同理，AE 曝光控制类型为手动时，配置自动曝光属性的参数无效。
- AE 曝光控制类型为手动时，若曝光参数设置超出最大（小）值，将使用 sensor 支持的最大（小）值代替。
- 无论是自动曝光还是手动曝光，曝光时间的单位为微秒(us)，曝光增益的单位为 10bit 精度的倍数，即 1024 代表 1 倍，2048 代表 2 倍等。
- 2 合 1WDR 模式下，设置增益分开配置，若 sensor 支持长短帧不同增益，可实现长短帧不同的 sensor 模拟增益、sensor 数字增益、WDR 增益，若 sensor 不支持长短帧不同增益，可实现长短帧不同的 WDR 增益，**通常建议长短帧使用相同增益设置。**

【相关主题】

[xmedia_ae_get_exposure_attr](#)

xmedia_ae_get_exposure_attr

【描述】

获取 AE 曝光属性。

【语法】

```
xmedia_s32 xmedia_ae_get_exposure_attr(xmedia_u32 pipe, xmedia\_ae\_exposure\_attr
*exposure_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
exposure_attr	AE 曝光属性结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_ae.h
- 库文件：libxmedia_isp.a、libxmedia_ae.a

【注意】

无

【举例】

无

【相关主题】

[xmedia_ae_set_exposure_attr](#)

[xmedia_ae_set_wdr_exposure_attr](#)

【描述】

设置 WDR 模式下的 AE 曝光属性。

【语法】

```
xmedia_s32 xmedia_ae_set_wdr_exposure_attr(xmedia_u32 pipe, const  
xmedia\_ae\_wdr\_exposure\_attr *wdr_exp_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
wdr_exp_attr	WDR 模式下的 AE 曝光属性结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_ae.h
- 库文件: libxmedia_isp.a、libxmedia_ae.a

【注意】

无

【举例】

无

【相关主题】

[xmedia_ae_get_wdr_exposure_attr](#)

xmedia_ae_get_wdr_exposure_attr

【描述】

获取 WDR 模式下的 AE 曝光属性。

【语法】

```
xmedia_s32 xmedia_ae_set_wdr_exposure_attr(xmedia_u32 pipe, const  
xmedia\_ae\_wdr\_exposure\_attr *wdr_exp_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
wdr_exp_attr	WDR 模式下的 AE 曝光属性结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_ae.h
- 库文件：libxmedia_ism.a、libxmedia_ae.a

【注意】

无

【举例】

无

【相关主题】

[xmedia_ae_set_wdr_exposure_attr](#)

xmedia_ae_set_route_attr

【描述】

设置 AE 曝光分配策略。

【语法】

```
xmedia_s32 xmedia_ae_set_route_attr(xmedia_u32 pipe, xmedia_isp_ae_route *ae_route_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ae_route_attr	AE 曝光分配策略结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

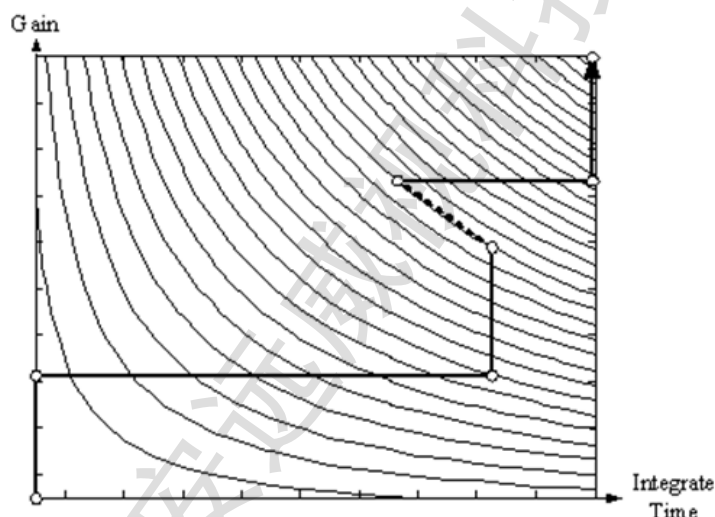
- 头文件: xmedia_ae.h
- 库文件: libxmedia_isp.a、libxmedia_ae.a

【注意】

- 此接口用于设定 AE 曝光分配路线，AE 计算得到的曝光量将按照设定的路线进行分配，用户可以根据自己的需求设定曝光优先、增益优先、光圈优先。
- AE 分配路线示意图如图 5-4 所示。AE 分配路线遵循以下限定：
 - 最大支持 16 个节点，每个节点有曝光时间、系统增益、光圈三个分量，系统增益包含模拟增益、数字增益、ISP 数字增益。
 - 节点中曝光时间的单位为 us，不能设置为 0，也不能设置太小导致实际对应的曝光行数为 0，否则可能产生异常。
 - 光圈分量仅支持 P-Iris，不支持 DC-Iris，因为 DC-Iris 无法精确控制，所以 DC-Iris 和手动光圈镜头光圈分量是无效的。即光圈类型为 DC-Iris 时，节点光圈分量不会对曝光量分配产生任何影响。
 - 节点的曝光量是曝光时间、增益和光圈的乘积，节点曝光量单调递增，后一个节点的曝光量大于或等于前一个节点的曝光量，第一个节点的曝光量最小，最后一个节点的曝光量最大。
 - 如果相邻节点的曝光量增加，那么应该有一个分量增加，其他分量固定，增加的分量决定该段路线的分配策略。例如增益分量增加，那么该段路线的分配策略是增益优先。
 - 不支持设置等曝光量节点。
 - 用户可以根据不同的场景设置不同的路线，分配路线支持动态切换。
 - 针对 DC-Iris 和手动光圈镜头，默认 AE 分配策略是首先分配曝光时间，其次分配增益。针对 P-Iris 镜头，默认 AE 分配策略是首先调节光圈，将光圈调至最大后调节曝光时间，最后再分配增益。如果当前曝光量不在用户设定的路线范围当中，按默认策略分配。

- 自动降帧时，若 sensor.c 里面设置了 AE route，则在切换后会采用 sensor.c 中的 AE route，否则采用已生效的 AE route。同时，最大曝光时间的改变会更新到 AE route 中。
- 线性模式与 WDR 模式切换时，若 sensor.c 里面设置了 AE route，则在切换后会采用 sensor.c 中的 route，否则采用默认的 AE route。
- 帧率或分辨率切换时，若用户设置的最大曝光目标时间大于切换后 1 帧所允许的最大曝光时间，那么分配路线的最大曝光时间会更新为切换后 1 帧所允许的最大曝光时间。
- 发生自动降帧、线性与 WDR 模式切换、帧率或分辨率切换、优先帧切换、增益分开配置切换、限制曝光时间或增益的最大最小值等情况时，实际生效的 AE route 可能与 MPI 设置的不一致，此时可以通过 `xmedia_ae_query_exposure_info` 获取实际生效的 AE route。

图5-4 AE 分配路线示意图



【举例】

无

【相关主题】

[xmedia_ae_get_route_attr](#)

`xmedia_ae_get_route_attr`

【描述】

获取 AE 曝光分配策略。

【语法】

```
xmedia_s32 xmedia_ae_get_route_attr(xmedia_u32 pipe, xmedia_isp_ae_route *ae_route_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ae_route_attr	AE 曝光分配策略结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_ae.h
- 库文件: libxmedia_isp.a、libxmedia_ae.a

【注意】

无

【举例】

无

【相关主题】

[xmedia_ae_set_route_attr](#)

[xmedia_ae_set_route_sf_attr](#)

【描述】

设置短帧 AE 曝光分配策略。

【语法】

```
xmedia_s32 xmedia_ae_set_route_sf_attr(xmedia_u32 pipe, xmedia_isp_ae_route
```

```
*ae_route_sf_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ae_route_sf_attr	短帧 AE 曝光分配策略结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_ae.h
- 库文件: libxmedia_isp.a、 libxmedia_ae.a

【注意】

无

【相关主题】

[xmedia_ae_get_route_sf_attr](#)

xmedia_ae_get_route_sf_attr

【描述】

获取短帧 AE 曝光分配策略。

【语法】

```
xmedia_s32 xmedia_ae_get_route_sf_attr(xmedia_u32 pipe, xmedia\_isp\_ae\_route  
*ae_route_sf_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ae_route_sf_attr	短帧 AE 曝光分配策略结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_ae.h
- 库文件: libxmedia_isp.a、libxmedia_ae.a

【注意】

无

【举例】

无

【相关主题】

[xmedia_ae_set_route_sf_attr](#)

xmedia_ae_query_exposure_info

【描述】

获取 ae 内部运行状态信息，包括全局 5 段直方图、1024 段直方图和平均亮度等统计信息，同时还可获取 AE 运行状态中的曝光时间、增益、曝光量和实际生效的 AE route 等信息。

【语法】

```
xmedia_s32 xmedia_ae_query_exposure_info(xmedia_u32 pipe, xmedia_ae_exp_info *exp_info);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
exp_info	AE 内部运行状态信息结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_ae.h
- 库文件: libxmedia_isp.a、libxmedia_ae.a

【注意】

- 获取的曝光时间以微秒(us)为单位，获取的 sensor 模拟增益、sensor 数字增益和 ISP 数字增益以倍数为单位，精度是 10bit。
- 获取的曝光量=(曝光时间 * 曝光增益)，未考虑光圈状态，其中曝光时间以行数为单位，曝光增益包括 sensor 模拟增益、sensor 数字增益和 ISP 数字增益，以倍数为单位，6bit 小数精度。若该值的精度不足以满足需求，可以根据上面高精度的曝光时间(us)和增益(10bit 小数精度)重新计算一个曝光量。
- 可通过查询 hist_error 来获取 AE 是否稳定的信息，若 hist_error 的绝对值小于曝光容忍偏差值，意味着当前 AE 不会动作。
- 通过该接口获取的 AE route 与 Proc 信息中的 AE route 都是实际生效的值，曝光时间以 us 为单位。
- 若用户使用非 XMEDIA_AE 算法，则该接口需要自己实现并且需要同步修改 PQTOOLS 的 xml 文件，具体修改方式请参考《图像质量调试工具使用指南》第 2.1.3.7 小节。
- 调用该接口需要确保系统已经运行起来并且统计信息已经生成；当系统刚运行起来第一帧，AE 内部还没有统计信息时，调用该接口统计信息会返回全 0，这时候系统内存是足够的。

【举例】

无

【相关主题】

无

xmedia_ae_set_ext_route_attr

【描述】

设置 AE 曝光分配扩展属性，支持分别设置 AE 分配策略中的 sensor 模拟增益，sensor 数字增益和 ISP 数字增益。

【语法】

```
xmedia_s32 xmedia_ae_set_ext_route_attr(xmedia_u32 pipe, const xmedia_isp_ae_ext_route  
*ae_ext_route_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ae_ext_route_attr	AE 曝光分配策略扩展属性结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_ae.h
- 库文件：libxmedia_isp.a、libxmedia_ae.a

【注意】

- 此接口用于设定 AE 曝光分配扩展属性，AE 计算得到的曝光量将按照设定的路线进行分配，用户可以根据自己的需求设定曝光时间优先、sensor 模拟增益优先、sensor 数字增益优先、ISP 数字增益优先和光圈优先。该接口可用于设置 WDR 模式下的曝光分配路线，减轻正常室内照度多帧合成 WDR 产生的工频闪现象，优化 WDR 模式图像效果。
- 曝光分配扩展属性是否生效可通过配置接口中的 `ae_route_ex_valid` 来实现。
`ae_route_ex_valid` 为 `XMEDIA_TRUE` 时使用扩展 AE route，否则使用 normal 模式 AE route。
- AE 扩展分配路线遵循以下限定：
 - 最大支持 16 个节点，每个节点有曝光时间、sensor 模拟增益、sensor 数字增益、ISP 数字增益和光圈五个分量。
 - 节点中曝光时间的单位为 us，不能设置为 0，也不能设置太小导致实际对应的曝光行数为 0，否则可能产生异常。
 - 光圈分量仅支持 P-Iris，不支持 DC-Iris，因为 DC-Iris 无法精确控制，所以 DC-Iris 和手动光圈镜头光圈分量是无效的。即光圈类型为 DC-Iris 时，节点光圈分量不会对曝光量分配产生任何影响。
 - 节点的曝光量是曝光时间、sensor 模拟增益、sensor 数字增益、ISP 数字增益和光圈的乘积，节点曝光量单调递增，后一个节点的曝光量大于或等于前一个节点的曝光量，第一个节点的曝光量最小，最后一个节点的曝光量最大。
 - 如果相邻节点的曝光量增加，那么应该有一个分量增加，其他分量固定，增加的分量决定该段路线的分配策略。例如 sensor 模拟增益分量增加，那么该段路线的分配策略是 sensor 模拟增益优先。
 - 不支持设置等曝光量节点。
 - 用户可以根据不同的场景设置不同的路线，分配路线支持动态切换。
 - 针对 DC-Iris 和手动光圈镜头，默认 AE 扩展分配策略是先分配曝光时间，再分配 sensor 模拟增益、sensor 数字增益，最后分配 ISP 数字增益。针对 P-Iris 镜头，默认 AE 扩展分配策略是先调节光圈，将光圈调至最大后调节曝光时间，最后再分配 sensor 模拟增益、sensor 数字增益和 ISP 数字增益。如果当前曝光量不在用户设定的路线范围当中，按默认策略分配。
 - 在线进行 DC-Iris 和 P-Iris 切换，扩展 AE route 会重置为与光圈类型相匹配的默认分配策略，用户可以根据需要在切换光圈类型时自行设置扩展 AE route。
 - 自动降帧时，最大曝光时间的改变会更新到分配路线中。

- 线性模式与 WDR 模式切换时，若 sensor.c 里面设置了扩展 AE route，且 sensor.c 中的扩展 AE route 标识符 ae_route_ex_valid 为 XMEDIA_TRUE，则在切换后会采用 sensor.c 中的扩展 AE route，否则采用默认的 AE route。
- 帧率或分辨率切换时，若用户设置的最大曝光目标时间大于切换后 1 帧所允许的最大曝光时间，那么分配路线的最大曝光时间会更新为切换后 1 帧所允许的最大曝光时间。
- 发生自动降帧、线性与 WDR 模式切换、帧率或分辨率切换、优先帧切换、增益分开配置切换、限制曝光时间或增益的最大最小值等情况时，实际生效的扩展 AE route 可能与 MPI 设置的不一致，此时可以通过 xmedia_ae_query_exposure_info 获取实际生效的扩展 AE route。

【举例】

无

【举例】

无

【相关主题】

[xmedia_ae_get_ext_route_attr](#)

xmedia_ae_get_ext_route_attr

【描述】

获取 AE 曝光分配策略扩展属性。

【语法】

```
xmedia_s32 xmedia_ae_get_ext_route_attr(xmedia_u32 pipe, const xmedia\_isp\_ae\_ext\_route
*ae_ext_route_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ae_ext_route_attr	AE 曝光分配策略扩展属性结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_ae.h
- 库文件: libxmedia_isp.a、 libxmedia_ae.a

【注意】

无

【相关主题】

[xmedia_ae_set_ext_route_attr](#)

xmedia_ae_set_ext_route_sf_attr

【描述】

设置短帧 AE 曝光分配扩展属性，支持分别设置 AE 分配策略中的 sensor 模拟增益，sensor 数字增益和 ISP 数字增益。

【语法】

```
xmedia_s32 xmedia_ae_set_ext_route_sf_attr(xmedia_u32 pipe, const xmedia\_isp\_ae\_ext\_route
*ae_ext_route_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ae_ext_route_attr	短帧 AE 曝光分配策略扩展属性结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_ae.h
- 库文件: libxmedia_esp.a、 libxmedia_ae.a

【注意】

无

【相关主题】

[xmedia_ae_get_ext_route_sf_attr](#)

xmedia_ae_get_ext_route_sf_attr

【描述】

获取短帧 AE 曝光分配策略扩展属性。

【语法】

```
xmedia_s32 xmedia_ae_get_ext_route_sf_attr(xmedia_u32 pipe, xmedia_esp_ae_ext_route
*ae_ext_route_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ae_ext_route_attr	短帧 AE 曝光分配策略扩展属性结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_ae.h
- 库文件: libxmedia_esp.a、libxmedia_ae.a

【注意】

无

【相关主题】

[xmedia_ae_set_ext_route_sf_attr](#)

xmedia_ae_set_smart_exposure_attr

【描述】

设置智能曝光参数。

【语法】

```
xmedia_s32 xmedia_ae_set_smart_exposure_attr(xmedia_u32 pipe, const
xmedia\_ae\_smart\_exposure\_attr *smart_exp_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
smart_exp_attr	AE 智能曝光参数结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_ae.h

•库文件: libxmedia_isp.a、libxmedia_ae.a

【注意】

无

【相关主题】

[xmedia_ae_get_smart_exposure_attr](#)

xmedia_ae_get_smart_exposure_attr

【描述】

获取智能曝光参数。

【语法】

```
xmedia_s32 xmedia_ae_set_smart_exposure_attr(xmedia_u32 pipe,  
xmedia\_ae\_smart\_exposure\_attr *smart_exp_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
smart_exp_attr	AE 智能曝光参数结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_ae.h
- 库文件: libxmedia_isp.a、libxmedia_ae.a

【注意】

无

【相关主题】

[xmedia_ae_set_smart_exposure_attr](#)

5.5 数据类型

5.5.1 Register

- [xmedia_sensor_register_ae_info](#): 定义 sensor 驱动与 AE 库之间接口结构体。
- [xmedia_sensor_register_ae_func](#): 定义 sensor 回调函数结构体。
- [xmedia_isp_ae_sensor_default](#): 定义 AE 算法库的初始化参数结构体。
- [xmedia_isp_ae_accuracy_type](#): 定义曝光时间、增益的精度类型的枚举。
- [xmedia_isp_ae_accuracy](#): 定义曝光时间、增益的精度度的结构体。

xmedia_sensor_register_ae_info

【说明】

定义 sensor 驱动与 AE 库之间接口结构体。

【定义】

```
typedef struct {  
    xmedia_u32 dev_id;  
    xmedia_sensor_register_ae_func func;  
} xmedia_sensor_register_ae_info;
```

【成员】

成员名称	描述
dev_id	sensor 设备号。
func	sensor 库提供的回调函数结构体。

【注意事项】

无。

【相关数据类型及接口】

xmedia_sensor_register_ae_func

xmedia_sensor_register_ae_func

【说明】

定义 sensor 回调函数结构体。

【定义】

```
typedef struct {  
    xmedia_s32 (*pfn_sensor_get_ae_default)(xmedia_u32 dev, xmedia_isp_ae_sensor_default  
*ae_sns_dft);  
    xmedia_s32 (*pfn_sensor_set_fps)(xmedia_u32 dev, xmedia_float fps,  
xmedia_isp_ae_sensor_default *ae_sns_dft);  
    xmedia_s32 (*pfn_sensor_set_slow_framerate)(xmedia_u32 dev, xmedia_u32 full_lines,  
xmedia_isp_ae_sensor_default *ae_sns_dft);  
    xmedia_s32 (*pfn_sensor_update_inttime)(xmedia_u32 dev, xmedia_u32 int_time);  
    xmedia_s32 (*pfn_sensor_calc_again)(xmedia_u32 dev, xmedia_u32 *again_lin, xmedia_u32  
*again_db);  
    xmedia_s32 (*pfn_sensor_calc_dgain)(xmedia_u32 dev, xmedia_u32 *dgain_lin, xmedia_u32  
*dgain_db);  
    xmedia_s32 (*pfn_sensor_update_gains)(xmedia_u32 dev, xmedia_u32 again, xmedia_u32  
dgain);  
    xmedia_s32 (*pfn_sensor_get_max_inttime)(xmedia_u32 dev, xmedia_sensor_ae_inttime_attr  
*inittime_attr);  
    xmedia_s32 (*pfn_sensor_set_ae_wdr_attr)(xmedia_u32 dev, xmedia_isp_ae_wdr_mode  
ae_wdr_mode);  
} xmedia_sensor_register_ae_func;
```

【成员】

成员名称	描述
pfn_sensor_get_ae_default	获取 ae 默认参数的回调函数指针。
pfn_sensor_set_fps	设置 sensor 帧率的回调函数指针。
pfn_sensor_set_slow_framerate	设置 sensor 降帧帧率的回调函数指针。
pfn_sensor_update_inttime	更新 sensor 曝光时间的回调函数指针。
pfn_sensor_calc_again	计算 sensor 模拟增益的回调函数指针。
pfn_sensor_calc_dgain	计算 sensor 数字增益的回调函数指针。

成员名称	描述
pfn_sensor_update_gains	更新 sensor 增益的回调函数指针。
pfn_sensor_get_max_inttime	获取 sensor 最大曝光时间的回调函数指针。
pfn_sensor_set_ae_wdr_attr	设置 AE WDR 属性的回调函数指针。

【注意事项】

- 如果回调函数指针不需要赋值，需要置为 NULL。
- 在 xmedia_isp_ae_sensor_default 中定义了曝光时间和增益的精度，pfn_sensor_update_inttime 和 pfn_sensor_update_gains 中设置的曝光时间和增益都是带精度的值，如何转换成 sensor 的配置值与 sensor 强相关，请参阅 sensor 手册。

【相关数据类型及接口】

xmedia_sensor_register_ae_func

xmedia_isp_ae_sensor_default

【说明】

定义 AE 算法库的初始化参数结构体。

【定义】

```
typedef struct {
    xmedia_u8          ae_compensation;
    xmedia_u32         lines_per_500ms;
    xmedia_u32         flicker_freq;
    xmedia_float       fps;
    xmedia_u32         hmax_times;
    xmedia_u32         init_exposure;
    xmedia_u32         init_ae_speed;
    xmedia_u32         init_ae_tolerance;
    xmedia_u32         full_lines_std;
    xmedia_u32         full_lines_max;
    xmedia_u32         full_lines;
    xmedia_u32         max_int_time;
    xmedia_u32         min_int_time;
    xmedia_u32         max_int_time_target;
    xmedia_u32         min_int_time_target;
}
```

```

xmedia_isp_ae_accuracy int_time_accu;
xmedia_u32             max_again;
xmedia_u32             min_again;
xmedia_u32             max_again_target;
xmedia_u32             min_again_target;
xmedia_isp_ae_accuracy again_accu;
xmedia_u32             max_dgain;
xmedia_u32             min_dgain;
xmedia_u32             max_dgain_target;
xmedia_u32             min_dgain_target;
xmedia_isp_ae_accuracy dgain_accu;
xmedia_u32             ispdgain_sep_thd;
xmedia_u32             max_ispdgain_target;
xmedia_u32             min_ispdgain_target;
xmedia_u32             ispdgain_shift;
xmedia_u32             max_int_time_step;
xmedia_u32             lf_max_short_time;
xmedia_u32             lf_min_exposure;
xmedia_isp_ae_route    ae_route_attr;
xmedia_bool            ae_route_ex_valid;
xmedia_isp_ae_ext_route ae_ext_route_attr;
xmedia_isp_ae_route    ae_route_sf_attr;
xmedia_isp_ae_ext_route ae_ext_route_sf_attr;
xmedia_u16             man_ratio_enable;
xmedia_u32             ratio[XMEDIA_ISP_AE_EXP_RATIO_NUM];
xmedia_isp_iris_type    iris_type;
xmedia_isp_piris_attr   piris_attr;
xmedia_isp_iris_f_no    max_iris_f_no;
xmedia_isp_iris_f_no    min_iris_f_no;
xmedia_isp_ae_strategy  ae_exp_mode;
xmedia_u16             iso_cal_coef;
xmedia_u8              ae_run_interval;
xmedia_u32             exp_ratio_max;
xmedia_u32             exp_ratio_min;
xmedia_bool            diff_gain_support;
} xmedia_isp_ae_sensor_default;

```

【成员】

成员名称	描述
ae_compensation	AE 亮度目标值，取值范围为[0,255]，建议用 0x38~0x40。

成员名称	描述
lines_per_500ms	每 500ms 的总行数。
flicker_freq	抗闪频率，数值为当前电源频率的 256 倍。
fps	基准帧率。
hmax_times	Sensor 读出一行的时间。
init_exposure	默认初始曝光量，等于曝光时间与曝光增益的乘积，其中曝光时间的单位为行数。 建议关注快速启动的产品形态根据常用场景配置一个合适的初始曝光量，以达到 AE 快速收敛。线性/WDR 模式切换时，也会采用该值作为切换后第一帧的曝光量。合理修改 sensor 初始化序列的曝光时间和增益，将曝光时间(行数)*增益(6bit 小数精度)计算得到曝光量，并赋值给该变量，可使得切换更加平滑。若不设置，该值默认为 0。
init_ae_speed	默认初始 AE 调节速度。 建议关注快速启动的产品形态可将该值配置为 128。若不设置，该值默认为 0。AE 算法内部会将该值钳位至[64, 255]。
init_ae_tolerance	默认初始 AE 曝光容忍偏差，AE 算法内部会将该值钳位至 (0, 255]。
full_lines_std	基准帧率时 1 帧的有效行数。
full_lines_max	sensor 降帧后 1 帧所能达到的最大行数，一般设为 sensor 所支持的最大行数。
full_lines	sensor 每帧实际生效的总行数。使用 AE 算法时，回调 pfn_sensor_set_fps 和 pfn_sensor_set_slow_framerate 时必须给该值赋值，用于返回每帧实际生效的总行数。
max_int_time	最大曝光时间，以行为单位。
min_int_time	最小曝光时间，以行为单位。
max_int_time_target	最大曝光时间目标值，以行为单位。
min_int_time_target	最小曝光时间目标值，以行为单位。

成员名称	描述
int_time_accu	曝光时间精度。
max_again	最大模拟增益，以倍为单位。
min_again	最小模拟增益，以倍为单位。
max_again_target	最大模拟增益目标值，以倍为单位。
min_again_target	最小模拟增益目标值，以倍为单位。
again_accu	模拟增益精度。
max_dgain	最大数字增益，以倍为单位。
min_dgain	最小数字增益，以倍为单位。
max_dgain_target	最大数字增益目标值，以倍为单位。
min_dgain_target	最小数字增益目标值，以倍为单位。
dgain_accu	数字增益精度。
ispgain_sep_thd	AINR 模式下，ispgain 在 FE 和 BE 进行分别配置的阈值，当 ispgain 小于等于 ispgain_sep_thd 时，生效点在 FE；当 ispgain 大于等于 ispgain_sep_thd 时，不超过阈值部分在 FE 生效，其余部分在 BE 生效。
max_ispgain_target	最大 ISP 数字增益目标值，以倍为单位。
min_ispgain_target	最小 ISP 数字增益目标值，以倍为单位。
ispgain_shift	ISP 数字增益精度。
max_int_time_step	WDR 模式下短帧曝光行数变化的最大步长。 暂未支持
lf_max_short_time	自动长帧模式下短帧最大曝光时间。 暂未支持
lf_min_exposure	自动长帧模式下强制输出长帧的最小曝光量。 暂未支持
ae_route_attr	AE 默认分配路线。
ae_route_ex_valid	AE 扩展分配路线是否生效开关，该值为 XMEDIA_TRUE 时使用扩展分配路线，否则使用普通 AE 分配路线。
ae_ext_route_attr	AE 默认扩展分配路线，合理配置可用于优化 WDR 模式图

成员名称	描述
	像效果。
ae_route_sf_attr	AE 短帧默认分配路线。
ae_ext_route_sf_attr	AE 短帧默认扩展分配路线，合理配置可用于优化 WDR 模式图像效果。
man_ratio_enable	多帧合成 WDR 手动曝光比使能，该值为 XMEDIA_TRUE 时，曝光比固定，不会根据场景自动更新。
ratio	多帧合成 WDR 相邻 2 帧默认曝光比。当 man_ratio_enable 为 XMEDIA_TRUE 时，采用 ratio [0] 作为 S/VS 默认曝光比，ratio [1] 作为 M/S 默认曝光比，ratio [2] 作为 L/M 默认曝光比。上述 VS, S, M, L 分别表示极短帧，短帧，中帧和长帧的曝光时间，2 帧合成时只有 VS 和 S；3 帧合成时采用 VS, S 和 M；4 帧合成时采用 VS, S, M 和 L。6bit 小数精度。 取值范围：[0x40, 0xFFF]
iris_type	默认镜头类型，DC-Iris 或 P-Iris。默认镜头类型与实际对接镜头类型不一致时，AI 算法无法正常工作。
piris_attr	P-Iris 参数，与具体镜头相关。只有默认镜头类型是 P-Iris 时，才需要用该结构体参数。
max_iris_f_no	P-Iris 可用的最大 F 值，与具体使用镜头相关，使用 P-Iris 时必须设置，用于限制实际生效的 max_iris_f_no 和 min_iris_f_no，避免光圈目标值落到不合理的范围导致 AE 分配异常。 取值范围为[ISP_IRIS_F_NO_32_0, ISP_IRIS_F_NO_1_0]。
min_iris_f_no	P-Iris 可用的最小 F 值，与具体使用镜头相关，使用 P-Iris 时必须设置，用于限制实际生效的 max_iris_f_no 和 min_iris_f_no，避免光圈目标值落到不合理的范围导致 AE 分配异常。 取值范围为[ISP_IRIS_F_NO_32_0, max_iris_f_no]。
ae_exp_mode	默认曝光策略，高光优先或低光优先。建议 FSWDR 模式

成员名称	描述
	设置为低光优先，线性模式及 BuiltIn WDR 设置为高光优先。若不设置该值，默认为高光优先。
iso_cal_coef	ISO 标定系数，用于保证拍照所需 DCF 信息中显示的 ISO 是标准的，8bit 精度。 取值范围：[0x0, 0xFFFF]，默认值为 0x100。
ae_run_interval	默认 AE 算法运行间隔，以帧为单位。若不设置，则默认 AE 每帧执行一次。 取值范围：(0x0, 0xFF]
exp_ratio_max	曝光比最大值
exp_ratio_min	曝光比最小值
diff_gain_support	长短帧增益分开配置功能开关

【注意事项】

- 线性/WDR 模式切换时，会回调 pfn_sensor_get_ae_default 函数更新 AE 相关默认参数。若 WDR 模式要使用 AE 扩展分配路线而线性模式不需要，建议在 pfn_sensor_get_ae_default 函数里面先对 AE 路线清零：ae_route_ex_valid = XMEDIA_FALSE，ae_route_attr.total_num = 0，ae_ext_route_attr.total_num = 0，然后视需要在 WDR 分支赋值。
- AE 算法采用 init_exposure 作为初始 5 帧的曝光量，可用于运动 DV 加速启动。建议关注快速启动的产品形态根据常用场景配置一个合适的初始曝光量，以达到 AE 快速收敛。WDR 模式该值对应的是长帧曝光量，WDR 模式若要快速启动，最好在 sensor.c 设置为固定曝光比，以减少曝光比调整的时间，待 AE 稳定后再根据需要设置为自动曝光比。若在 sensor.c 未给 init_exposure 赋值或将 init_exposure 赋值为 0，则 AE 算法内部按起始曝光量为 1024 开始计算。WDR 模式切换时，AE 算法内部会计算切换曝光量以保证切换平滑，如果希望在 WDR 模式切换时 init_exposure 生效，可以通过配置 init_ae_speed 为 0xFFFFFFFF 实现。
- 对于特定 Sensor，如 OV2718 DCG 模式，内部仅支持固定曝光比，需要配置 man_ratio_enable = XMEDIA_TRUE，并把 ratio 配置为 Sensor 支持的固定曝光比，如果使用自动曝光比或手动配置为不支持的曝光比，会使图像效果不正常。
- max_inttime_step 自动长帧模式下普通模式和长帧模式之前切换过程中，对短帧曝光时间减小的最大调整步长，以行数为单位。仅在自动长帧模式下有效。此参

数针对短帧曝光时间减小过快会出现坏帧的 sensor 开放，一般 sensor 可以设定为较大值，则可以受此参数限制。

- If_max_short_time 为自动长帧模式下短帧曝光时间的最大值，如果此参数设置过小会导致自动长帧模式下亮区噪声表现变差。

【相关数据类型及接口】

xmedia_sensor_register_ae_func

xmedia_isp_ae_accuracy_type

【说明】

定义曝光时间、增益的精度类型的枚举。

【定义】

```
typedef enum {
    XMEDIA_ISP_AE_ACCURACY_TYPE_DB = 0,
    XMEDIA_ISP_AE_ACCURACY_TYPE_LINEAR,
    XMEDIA_ISP_AE_ACCURACY_TYPE_TABLE,
    XMEDIA_ISP_AE_ACCURACY_TYPE_MAX,
} xmedia_isp_ae_accuracy_type;
```

【成员】

成员名称	描述
XMEDIA_ISP_AE_ACCURACY_TYPE_DB	DB 精度类型。
XMEDIA_ISP_AE_ACCURACY_TYPE_LINEAR	线性精度类型。
XMEDIA_ISP_AE_ACCURACY_TYPE_TABLE	表格类型。

【注意事项】

无

【相关数据类型及接口】

xmedia_isp_ae_accuracy

xmedia_isp_ae_accuracy

【说明】

定义曝光时间、增益的精度结构体。

【定义】

```
typedef struct {  
    xmedia_esp_ae_accuracy_type accu_type;  
    xmedia_float                accuracy;  
    xmedia_float                offset;  
} xmedia_esp_ae_accuracy;
```

【成员】

成员名称	描述
accu_type	精度类型，包括线性类型、DB 类型和 TABLE 类型。
accuracy	精度值。
offset	曝光时间的偏移量，支持正负偏移量设置，以行为单位，该值配置与 sensor 强相关。

【注意事项】

- 大部分增益的精度均可归结为线性精度、DB 精度和 TABLE 精度这几类。
- 线性精度指的是增益的倍数均匀增加。例如 sensor 能支持的 again 为 1 倍，1(1+1/16)倍，1(1+2/16)倍.....这种情况下，精度类型可以设定为 XMEDIA_ISP_AE_ACCURACY_TYPE_LINEAR，精度值设置为 0.0625，在这种精度下，again 为 32 则表示 $32 \times 0.0625 = 2$ 倍增益。
- DB 精度指的是增益的倍数以倍增的形式增加。例如 sensor 的芯片手册说明能支持的 again 为 0db，0.3db，0.6db.....这种情况下，精度类型可以设定为 XMEDIA_ISP_AE_ACCURACY_TYPE_DB，精度值设置为 0.3，在这种精度下，again 为 80 则表示 $80 \times 0.3\text{db} = 24\text{db}$ ，24db 是 16 倍增益。再例如 sensor 能支持的 again 为 1 倍，2 倍，4 倍，8 倍.....这种情况下，对应为 db 则为 0db，6db，12db，18db，所以精度类型设定为 XMEDIA_ISP_AE_ACCURACY_TYPE_DB，精度值设置为 6。由于 AE 算法内部实现线性转 DB 时计算精度存在误差，当 DB 精度较高(小于 1)时，建议采用 TABLE 精度来实现。
- TABLE 精度指增益的倍数是通过查表的形式获得，表格统一使用 10bit 精度，即 1024 表示 1 倍增益。当某些 sensor 的增益值的增加规律非线性或者 DB 精度较

高时，可以使用 TABLE 方式，AE 算法计算出需要的模拟增益/数字增益的数值，用查询 sensor 增益表格中最接近的值作为数字增益/模拟增益的值。

- 使用 TABLE 模式时，需要相应的初始化回调结构体 `xmedia_sensor_register_ae_func` 中的回调函数。
- 曝光时间的精度通常都是线性的，都是以行为单位。若曝光时间为线性精度，精度 `accuracy` 为大于 1 的整数，则意味着 AE 计算出来的曝光行数只能为最小曝光时间加上 `accuracy` 的整数倍。该功能可用于计算某些 sensor WDR 模式下的长/短帧曝光时间，如某些 sensor 曝光行数必须为 $2n+1$ ，则可以配置最小曝光时间为 1 或 3，配置 `accuracy` 为 2。
- 曝光时间与 sensor 增益保持线性关系是 AE 曝光量分配的前提，即认为曝光量一定的情况下，曝光时间和 sensor 增益可以相互转换而保持图像亮度基本不变。比如说曝光量为 4096，那么分配为曝光时间(2)*增益(2048)，也可以分配为曝光时间(4)*增益(1024)，这 2 种情况下图像亮度应该基本不变。若不满足这种线性关系，在高亮环境下，由于曝光时间很短，1 行曝光时间的变化也容易使画面发生闪烁。某些 sensor，如 Panasonic MN34220 采用 1080p@30fps 的初始化序列配置时，曝光时间必须偏移 0.8018 行之后才与 sensor 增益有线性关系，因此增加了 `offset` 这个变量，它为 float 型变量，以行为单位，将 `offset` 设置成 0.8018，AE 算法内部即可完成偏移处理。一般 sensor 的曝光时间和增益不存在这种偏移关系，需将该值配为 0。
- 规定这 3 种精度类型，可以以统一的接口形式对接绝大部分 sensor。

【相关数据类型及接口】

`xmedia_sensor_register_ae_func`

5.5.2 AE

- `xmedia_ae_prior_frame`: 定义 WDR 模式下的优先帧。
- `xmedia_ae_range`: 定义曝光时间或增益的最大值和最小值。
- `xmedia_isp_ae_strategy`: 定义 AE 曝光策略模式。
- `xmedia_ae_mode`: 定义自动曝光的模式。
- `xmedia_ae_anti_flicker_mode`: 定义抗闪模式。
- `xmedia_ae_anti_flicker`: 定义抗闪属性。
- `xmedia_ae_sub_flicker`: 定义 ISP 图像亚抗闪属性。
- `xmedia_ae_delay`: 定义 AE 延时属性。
- `xmedia_isp_wdr_mode`: 定义 ISP WDR 运行模式。

- [xmedia_ae_auto_attr](#): 定义自动曝光属性。
- [xmedia_ae_manual_attr](#): 定义手动曝光属性。
- [xmedia_ae_exposure_attr](#): 定义 ISP 曝光属性。
- [xmedia_ae_wdr_exposure_attr](#): 定义 WDR 模式下的曝光属性。
- [xmedia_isp_ae_route_node](#): 定义 AE 分配路线节点属性。
- [xmedia_isp_ae_route](#): 定义 AE 曝光分配策略属性。
- [xmedia_ae_exp_info](#): 定义 ISP 曝光内部状态信息。
- [xmedia_isp_ae_ext_route_node](#): 定义 AE 扩展分配路线节点属性。
- [xmedia_isp_ae_ext_route](#): 定义 AE 曝光分配策略扩展属性。
- [xmedia_ae_smart_exposure_attr](#): 定义智能模式下的曝光属性。

xmedia_ae_prior_frame

【说明】

定义 WDR 模式下的帧优先级。

【定义】

```
typedef enum {
    XMEDIA_LONG_FRAME = 0,
    XMEDIA_SHORT_FRAME = 1,
    XMEDIA_PRIOR_FRAME_MAX
} xmedia_ae_prior_frame;
```

【成员】

成员名称	描述
XMEDIA_LONG_FRAME	长帧。
XMEDIA_SHORT_FRAME	短帧。

【注意事项】

暂未开放该功能，固定以长帧为基准帧进行 AE 计算

【相关数据类型及接口】

[xmedia_ae_set_exposure_attr](#)
[xmedia_ae_get_exposure_attr](#)

xmedia_ae_range

【说明】

定义曝光时间或增益的最大值和最小值。

【定义】

```
typedef struct {  
    xmedia_u32 max;  
    xmedia_u32 min;  
} xmedia_ae_range;
```

【成员】

成员名称	描述
max	最大值。
min	最小值。

【注意事项】

最小值不能大于最大值。

【相关数据类型及接口】

无

xmedia_isp_ae_strategy

【说明】

定义 AE 曝光策略模式。

【定义】

```
typedef enum {  
    XMEDIA_ISP_AE_STRATEGY_EXP_HIGHLIGHT_PRIOR = 0,  
    XMEDIA_ISP_AE_STRATEGY_EXP_LOWLIGHT_PRIOR = 1,  
    XMEDIA_ISP_AE_STRATEGY_MAX  
} xmedia_isp_ae_strategy;
```

【成员】

成员名称	描述
XMEDIA_ISP_AE_STRATEGY_EXP_HIGHLIGHT_PRIOR	高光优先曝光模式。
XMEDIA_ISP_AE_STRATEGY_EXP_LOWLIGHT_PRIOR	低光优先曝光模式。

【注意事项】

线性模式下，XMEDIA_AE 算法默认曝光策略为高光优先，尽量避免画面过曝，但在宽动态场景时，暗处亮度较低。此时若主要关注暗处区域，可采用低光优先模式，但亮处容易过曝。

【相关数据类型及接口】

无

xmedia_ae_mode

【说明】

定义自动曝光的模式。

【定义】

```
typedef enum {
    XMEDIA_AE_MODE_SLOW_SHUTTER    = 0,
    XMEDIA_XMEDIA_AE_MODE_FIX_FRAME_RATE = 1,
    XMEDIA_AE_MODE_MAX
} xmedia_ae_mode;
```

【成员】

成员名称	描述
XMEDIA_AE_MODE_SLOW_SHUTTER	自动降帧模式，即 SLOW_SHUTTER 模式。
XMEDIA_AE_MODE_FIX_FRAME_RATE	固定帧率模式。

【注意事项】

- 自动降帧模式是指自动曝光调节时会优先增大曝光时间来尽量减小增益。当系统增益达到用户设置的最大值时，自动曝光调节会逐渐降帧率并延长曝光时间，持续到曝光时间等于自动曝光的最大时间为止。低照度环境下噪声较小但帧率会降低。
- 固定帧率模式是指自动曝光调节时保持帧率不变，低照度环境下噪声会较大。

【相关数据类型及接口】

无

xmedia_ae_anti_flicker_mode

【说明】

定义抗闪模式。

【定义】

```
typedef enum {
    XMEDIA_AE_ANTIFLICKER_NORMAL_MODE = 0x0,
    XMEDIA_AE_ANTIFLICKER_AUTO_MODE   = 0x1,
    XMEDIA_AE_ANTIFLICKER_MODE_MAX
} xmedia_ae_anti_flicker_mode;
```

【成员】

成员名称	描述
XMEDIA_AE_ANTIFLICKER_NORMAL_MODE	普通抗闪模式。
XMEDIA_AE_ANTIFLICKER_AUTO_MODE	自动抗闪模式。

【注意事项】

- XMEDIA_AE_ANTIFLICKER_NORMAL_MODE 为强制抗闪模式，曝光时间可以根据亮度进行调节，最小曝光时间固定为 1/120 sec（60Hz）或 1/100 sec(50Hz)，不受曝光时间最小值的限制。
 - 有灯光的环境：曝光时间可与光源频率相匹配，能够防止图像闪烁。
 - 高亮度的环境：亮度越高，要求曝光时间就最短。而强制抗闪模式的最小曝光时间不能匹配光源频率，产生过曝。
- XMEDIA_AE_ANTIFLICKER_AUTO_MODE 为非强制抗闪模式，曝光时间可以根据亮度进行调节，最小曝光时间可达到 sensor 的最小曝光时间。与强制抗闪模式的差别主要在高亮度的环境体现。

- 高亮度的环境：最小曝光时间可以达到 sensor 的最小曝光时间，能够有效抑制过曝，但此时抗闪失效。

【相关数据类型及接口】

无

xmedia_ae_anti_flicker

【说明】

定义抗闪属性。

【定义】

```
typedef struct ISP_ANTIFLICKER_S {
    xmedia_bool    enable;
    xmedia_u8      frequency;
    xmedia_ae_anti_flicker_mode mode;
} xmedia_ae_anti_flicker;
```

【成员】

成员名称	描述
enable	enable 为 XMEDIA_TRUE 时使能图像抗闪，为 XMEDIA_FALSE 时不使能图像抗闪。
frequency	抗闪频率值。 取值范围：[0x0, 0xFF]，默认值为 0x0。
mode	抗闪模式，普通抗闪或自动抗闪。

【注意事项】

- 抗闪开启后曝光时间会受到最大/最小曝光时间的限制，若最小抗闪时间大于最大曝光时间，那么开启抗闪后曝光时间将被限制为最大曝光时间。
- 开启抗闪时，AERoute 中各节点的 Again、DGain、ISPDGain 参数至少有一个需小于对应参数范围的上限，避免特定场景下无法进行补偿导致图像闪烁。

【相关数据类型及接口】

xmedia_ae_anti_flicker_mode

xmedia_ae_sub_flicker

【说明】

定义 ISP 图像亚抗闪属性。

【定义】

```
typedef struct {
    xmedia_bool enable;
    xmedia_u8    luma_diff;
} xmedia_ae_sub_flicker;
```

【成员】

成员名称	描述
enable	enable 为 XMEDIA_TRUE 时使能图像亚抗闪功能，为 XMEDIA_FALSE 时不使能图像亚抗闪。
luma_diff	抗闪程度设置，范围 [0x0, 0x64]。亚抗闪功能生效时，该值越大越接近于抗闪。

【注意事项】

- 强制抗闪模式时，最小曝光时间固定为 1/120 sec (60Hz) 或 1/100 sec(50Hz)，在某些场景（如室内对准室外窗户的背光场景）画面可能会过曝严重，但不抗闪画面工频闪又比较厉害。在这种情况下引入亚抗闪模式是为了取得过曝与闪烁之间的平衡。在强制抗闪模式下，当亚抗闪功能生效时，若画面亮度小于 (compensation + luma_diff)，那么曝光时间仍会固定为最小抗闪时间 1/120 sec (60Hz) 或 1/100 sec(50Hz)以防止图像闪烁。若画面亮度大于(compensation + luma_diff)，则取消抗闪，调整画面目标亮度为(compensation + luma_diff)，通过引入一定程度的画面闪烁来避免画面过曝严重。注意：非背光场景 luma_diff 较小时（如小于 10）可能会导致曝光时间在最小抗闪点左右来回变化，导致画面轻微闪烁，建议设置该值不小于 20。实际上，若 luma_diff 设置较小，背光场景下相当于取消了抗闪功能，与自动抗闪模式下的效果相近，达不到亚抗闪的目的，因此该值只有设置一个稍大些的值才有意义。
- 只有在满足打开抗闪、强制抗闪模式且抗闪频率值不等于 0 的前提下，亚抗闪功能才能生效。
- 目前亚抗闪功能暂未开放，相关参数调试无效果。

【相关数据类型及接口】

无

xmedia_ae_delay

【说明】

定义 AE 延时属性。

【定义】

```
typedef struct {  
    xmedia_u16 black_delay_frame;  
    xmedia_u16 white_delay_frame;  
} xmedia_ae_delay;
```

【成员】

成员名称	描述
black_delay_frame	图像亮度小于目标亮度时间超过 black_delay_frame 帧时，AE 开始调节。
white_delay_frame	图像亮度大于目标亮度时间超过 white_delay_frame 帧时，AE 开始调节。

【注意事项】

- black_delay_frame/white_delay_frame 设置越大，在 AE 调节步长相同的情况下，调节至目标亮度需要越长时间。
- 人眼对于过曝比较敏感，建议 white_delay_frame 要设置得比 black_delay_frame 小一些。
- 固定降帧时，若帧率较低，black_delay_frame/white_delay_frame 要做出相应调整，建议不要设置过大，否则 AE 收敛时间会较长。

【相关数据类型及接口】

无

xmedia_isp_wdr_mode

【说明】

定义 ISP WDR 运行模式。

【定义】

```
typedef enum {  
    XMEDIA_ISP_WDR_MODE_NORMAL          = 0x0,  
    XMEDIA_ISP_WDR_MODE_LONG_FRAME      = 0x1,  
    XMEDIA_ISP_WDR_MODE_AUTO_LONG_FRAME = 0x2,  
    XMEDIA_ISP_WDR_MODE_MAX  
} xmedia_isp_wdr_mode;
```

【成员】

成员名称	描述
XMEDIA_ISP_WDR_MODE_NORMAL	正常 WDR 模式，此时 AE 和合成模块按照自动/手动曝光比进行工作。
XMEDIA_ISP_WDR_MODE_LONG_FRAME	长帧模式，此时 AE 将短帧曝光时间设置为最小值，长帧曝光时间接近 1 帧所允许的最大值，合成模块只输出长帧数据，可用于优化弱宽动态场景或低照度时的图像质量。
XMEDIA_ISP_WDR_MODE_AUTO_LONG_FRAME	自动长帧模式，此时 AE 短帧曝光时间不会超过 sensor 限制的长帧模式下短帧最大曝光时间，在曝光量超过 sensor 设置的阈值时，自动切换为长帧模式，合成模块只输出长帧数据。

【注意事项】

- 长帧模式仅在行模式 WDR 长帧优先模式下有效，自动长帧模式仅在 2to1 行模式 WDR 长帧优先模式下有效，并且为了保证图像质量，WDR 模块的运动检测及长短帧融合阈值均为自动配置，不支持手动配置。在线将模式切换到 WDR 模式，默认为正常 WDR 模式。
- 自动长帧模式下，手动曝光比生效，如果手动曝光比大于 1，即使曝光量大于设置的阈值也不会进入长帧模式；如果手动曝光比为 1，则会自动进入长帧模式。

- 长帧模式和自动长帧模式仅支持在优先帧为长帧且增益分开配置不使能的情况下使用。
- **目前暂未开放长帧/自动长帧模式，相关参数调试无效果。**

【相关数据类型及接口】

无

xmedia_ae_auto_attr

【说明】

定义自动曝光属性。

【定义】

```
typedef struct {
    xmedia_ae_range    exp_time_range;
    xmedia_ae_range    again_range;
    xmedia_ae_range    dgain_range;
    xmedia_ae_range    isp_dgain_range;
    xmedia_ae_range    sys_gain_range;
    xmedia_u32          gain_thres_hold;
    xmedia_u8           speed;
    xmedia_u16          black_speed_bias;
    xmedia_u8           tolerance;
    xmedia_u8           iso_num;
    xmedia_u32          iso_level[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8           compensation[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16          ev_bias[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_isp_ae_strategy ae_strategy_mode;
    xmedia_u16          hist_ratio_slope[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8           max_hist_offset[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_ae_mode       ae_mode;
    xmedia_ae_anti_flicker anti_flicker;
    xmedia_ae_sub_flicker sub_flicker;
    xmedia_ae_delay       ae_delay_attr;
    xmedia_bool          manual_exp_value;
    xmedia_u32           exp_value;
    xmedia_isp_wdr_mode   wdr_mode;
    xmedia_bool          wdr_quick;
    xmedia_u16           iso_cal_coef;
} xmedia_ae_auto_attr;
```

【成员】

成员名称	描述
exp_time_range	曝光时间范围，设置最大值和最小值，以微秒(us)为单位。 取值范围：[0x0, 0xFFFFFFFF]，具体范围与 sensor 相关。
again_range	Sensor 模拟增益范围，设置最大值和最小值，10bit 小数精度。 取值范围：[0x400, 0xFFFFFFFF]，具体范围与 sensor 相关。
dgain_range	Sensor 数字增益范围，设置最大值和最小值，10bit 小数精度。 取值范围：[0x400, 0xFFFFFFFF]，具体范围与 sensor 相关。
isp_dgain_range	ISP 数字增益范围，设置最大值和最小值，10bit 小数精度。 取值范围：[0x400, 0x40000]。
sys_gain_range	系统增益范围，设置最大值和最小值，10bit 小数精度。 取值范围：[0x400, 0xFFFFFFFF]，具体范围与 sensor 相关。
gain_thres_hold	自动降帧时的系统增益门限值，10bit 小数精度。 取值范围：[0x400, 0xFFFFFFFF]。
speed	自动曝光调整时的速度。 取值范围：[0x0, 0xFF]，默认值为 0x40。
black_speed_bias	画面由暗到亮 AE 调节速度的偏差值，该值越大，画面从暗到亮的速度越快。 取值范围：[0x0, 0xFFFF]，默认值为 0x90。
tolerance	自动曝光调整时对画面亮度的容忍偏差。 取值范围：[0x0, 0xFF]，默认值为 0x2。
iso_num	ISO 节点数量。
iso_level	ISO 节点数值。
compensation	自动曝光调整时的目标亮度。

	取值范围：[0x0, 0xFF], 默认值为 0x38。
ev_bias	自动曝光调整时的目标亮度调整值，10bit 小数精度。 取值范围：[0x0, 0xFFFF], 默认值为 0x400。
ae_strategy_mode	自动曝光策略，高光优先或低光优先。
hist_ratio_slope	感兴趣区域的权重。 取值范围：[0x0, 0xFFFF], 默认值为 0。
max_hist_offset	感兴趣区域对统计亮度影响上限值。 取值范围：[0x0, 0xFF], 默认值为 0x40。
ae_mode	自动曝光模式，自动降帧模式或固定帧率模式。
anti_flicker	抗闪属性设置。默认抗闪不使能。
sub_flicker	亚抗闪属性设置。默认亚抗闪不使能。 暂未支持
ae_delay_attr	延时属性设置。默认 black_delay_frame =8, white_delay_frame=0。
manual_exp_value	手动曝光量使能，该值为 XMEDIA_TRUE 时，AE 算法采用 exp_value 作为当前曝光量进行曝光时间和增益等的分配，为 XMEDIA_FALSE 时采用自动计算的曝光量进行分配。默认为 XMEDIA_FALSE。
exp_value	手动曝光量值，等于曝光时间*系统增益，其中曝光时间的单位为微秒(us)。 取值范围：(0x0, 0xFFFFFFFF]。
wdr_mode	WDR 运行模式。 暂未支持 默认为 XMEDIA_ISP_WDR_MODE_NORMAL。
wdr_quick	WDR 模式下，AE 算法从稳定状态（亮度误差小于等于容忍偏差值 tolerance）重新调整时，默认前 50 帧调整会进行时域滤波，以调整更加平滑。该值为 XMEDIA_TRUE 时，取消 50 帧时域滤波，使 AE 收敛速度更快。默认为 XMEDIA_FALSE。 暂未支持
iso_cal_coef	ISO 标定系数，用于保证拍照所需 DCF 信息中显示的 ISO 是标准的，8bit 精度。

	取值范围: [0x0, 0xFFFF], 默认值为 0x100。
--	----------------------------------

【注意事项】

- 自动模式下，更改手动曝光属性的值不会生效。
- 自动曝光的最大最小时间及增益。

可根据不同的场景对曝光时间及增益进行限定，如有高速运动物体场景可限定最大曝光时间值为较小值，这样可减轻运动物体拖影现象。

注意：

为了防止图像过曝区域颜色不饱和，驱动内部会根据 sensor 黑电平自动修正最小的 ISPDGain 值，因此 AE 在自动模式下，即使限制 ISPDGain 最大值为 1024，ISPDGain 实际生效值仍会大于 1024。

- 自动曝光的系统增益

非扩展路径下，若(sensor 模拟增益最小值*sensor 数字增益最小值*ISP 数字增益最小值)小于系统增益最小值，则 AE 算法内部计算时，最小增益会受到各分量最小增益限制。若(sensor 模拟增益最大值*sensor 数字增益最大值*ISP 数字增益最大值)大于系统增益最大值，则 AE 算法内部计算时，最大增益会受各分量最小增益限制。推荐通过设置系统增益的最大、最小值进行增益限制，分别限制 sensor 模拟增益、sensor 数字增益和 ISP 数字增益时，若把较高精度的 ISP 数字增益限制到 1 倍，容易导致闪烁。

- 自动降帧时的系统增益门限值

在 SLOW_SHUTTER 模式下，当系统增益达到所设置的门限值时，系统将自动进入 SLOW_SHUTTER 模式。

- speed 用于设定自动曝光时的收敛速度，该值越大曝光的收敛速度越快，但也会导致收敛过程中出现反复震荡。
- black_speed_bias 用于设定画面由暗到亮 AE 调节速度的偏差值。默认 speed 下，画面从亮到暗的速度会快于画面从暗到亮的速度，若想双方向的 AE 调节速度差不多，可以调高 black_speed_bias。
- 亮度补偿属性 compensation 用于调节曝光的目标亮度。曝光亮度补偿值越大则图像亮度越高。
- 曝光偏差属性 ev_bias 用于在特殊场景下对画面的目标亮度进行微调，也可认为是更高精度的亮度补偿值，通过调节该值来改变画面目标亮度，真实生效的 AE 目标亮度为 $\text{compensation} * \text{ev_bias} / 1024$ 。compensation 不变时，该值越大则图像亮度越高。

- 曝光容忍偏差属性 tolerance 用于调节曝光对环境的灵敏度，曝光容忍偏差值越大则曝光越不敏感，且可能导致同一目标亮度值多次调节得到的亮度差异越大，所以该属性推荐不能设定过大。尤其当目标亮度较低时，tolerance 较大，最终画面收敛亮度可能会有明显差异。通过改变 ev_bias 对画面目标亮度进行微调时，若 tolerance 较大，过小的 ev_bias 变化量可能会看不出调整效果。
- 曝光策略属性 ae_strategy_mode 用于选择对高光优先或低光优先的曝光策略。高光优先意味着对高光敏感，尽量避免画面过曝。低光优先意味着对低光敏感，尽量看清楚暗处区域，不管画面是否过曝。默认的曝光策略是高光优先，用户可根据场景需要进行调整。
- hist_ratio_slope 用于设定感兴趣区域的权重。若是高光优先曝光模式，则该值设置的是高光区域的权重，该值越大，则意味着对高光区域越敏感。反之，若是低光优先模式，则该值设置的是低光区域的权重，该值越大，则意味着对暗处区域越敏感。高光优先曝光模式下，如果该值过大，只要画面中有一小部分过曝区域，AE 算法就会认为画面亮度远大于目标亮度，这样曝光量就会不断往下调，导致画面变得很黑，同时画面中的过曝区域也在快速变小，达到某个程度的话 AE 算法又会认为画面亮度比目标亮度小了，曝光量又会往上调，如此反复出现来回调的现象，看起来像是画面闪烁。类似的，在低光优先模式下，如果 AE 目标亮度设置得比较低，若画面中有黑色物体来回移动，对画面亮度也会有较大影响。
- max_hist_offset 用于限制 hist_ratio_slope 对感兴趣区域统计权重提高的上限值，若该值为 0，无论 hist_ratio_slope 多大，也不会对高光或低光区域做特殊处理，此时的统计平均值就是原始值。通过合理设置该值，可以保证任何场景 AE 稳定后画面平均亮度都在一定范围内，在高光优先曝光模式下，如果该值设置较大，在对比度稍高一些的场景，如晴天室外场景，有天空有树木，则可能导致整体画面亮度偏低，因为此时优先保证了亮区天空的效果，通过限制该值从而限制对亮区的倾斜程度可以解决该问题。
- 曝光策略切换时，最好同时更新 hist_ratio_slope 和 max_hist_offset 的值，否则这 2 个值会采用默认配置，效果可能与预期不符。
- 在做强光抑制方案时，例如在 RGBIR 模式下，建议设置成高光优先曝光，通过降低 AE 目标亮度，同时适当设置 hist_ratio_slope 和 max_hist_offset 为较大值来抑制强光，暗区则可以通过使能 DRC 来看清楚。低光优先则可以用于实现非指定区域的背光补偿。
- AE 曝光控制类型为自动时，可设定曝光模式 ae_mode。该值可设定为慢快门模式 (SLOW_SHUTTER) 或固定帧率模式。慢快门模式通常用于低照度场景下进行自动降帧，以减少画面噪声。

- 抗闪属性结构体 anti_flicker 可用于设定抗闪使能，抗闪频率和抗闪模式等属性。
WDR 模式下仅长帧抗闪生效。
- AE 曝光控制类型为自动时，可设定 AE 延时生效属性结构体 ae_delay_attr。该值的合理设置可提高画面亮度的稳定性，防止快速运动物体经过导致画面亮度发生变化。低码率设置时可适当增加该值，以避免 AE 调节时出现块效应。
- XMEDIA 用户可以通过将 manual_exp_value 置为 XMEDIA_TRUE，手动设置曝光量 exp_value 来屏蔽 AE 算法的曝光量调整部分，只用到曝光量分配部分。
exp_value 的值会受到最大、最小曝光量的限制。最大、最小曝光量分别对应最大、最小曝光时间与增益的乘积。
- WDR 运行模式改变时，需要相应修改 sensor.c 中的以下回调函数：
pfn_sensor_get_ae_default, pfn_sensor_get_max_inttime,
pfn_sensor_set_fps, 以保证长、短帧曝光时间在合理范围内。正常 WDR 和长帧模式切换时，会回调 cmos_get_ae_default，更新曝光时间最大/最小值，增益最大/最小值，光圈最大/最小值和 AE compensation 等参数，同时，AE 内部会将 hist_ratio_slope 和 max_hist_offset 更新为算法默认值。需要注意的是，此时曝光比相关设置不会更新，仍会保持上一次的值。

注意：平滑切换的前提是切换前/后短帧曝光量与长帧曝光量相等。当曝光比较大时，由于正常 WDR 模式会限制增益，而长帧模式为了保证低照度效果会放开增益限制，可能会导致切换前/后短帧曝光量与长帧曝光量不等，这样切换就会有几帧突然之间变亮或者变暗，属于正常现象。若要避免这种现象，可以在切换完成之后再放开增益限制。

【相关数据类型及接口】

[xmedia_ae_set_exposure_attr](#)

xmedia_ae_manual_attr

【说明】

定义手动曝光属性。

【定义】

```
typedef struct {
    xmedia_video_operation_mode exp_time_op_type;
    xmedia_video_operation_mode again_op_type;
    xmedia_video_operation_mode dgain_op_type;
    xmedia_video_operation_mode isp_dgain_op_type;
    xmedia_u32 exp_time;
    xmedia_u32 again;
```

```

xmedia_u32      dgain;
xmedia_u32      isp_dgain;
} xmedia_ae_manual_attr;

```

【成员】

成员名称	描述
exp_time_op_type	手动曝光时间使能, 默认值为 XMEDIA_VIDEO_OPERATION_MODE_AUTO
again_op_type	手动 sensor 模拟增益使能, 默认值为 XMEDIA_VIDEO_OPERATION_MODE_AUTO
dgain_op_type	手动 sensor 数字增益使能, 默认值为 XMEDIA_VIDEO_OPERATION_MODE_AUTO
isp_dgain_op_type	手动 ISP 数字增益使能, 默认值为 XMEDIA_VIDEO_OPERATION_MODE_AUTO
exp_time	手动曝光时间, 以微秒(us)为单位, 默认值为 0x4000。 取值范围: [0x0, 0xFFFFFFFF], 具体范围与 sensor 相关。
again	手动 sensor 模拟增益, 10bit 小数精度, 默认值为 0x400。 取值范围: [0x400, 0xFFFFFFFF], 具体范围与 sensor 相关。
dgain	手动 sensor 数字增益, 10bit 小数精度, 默认值为 0x400。 取值范围: [0x400, 0xFFFFFFFF], 具体范围与 sensor 相关。
isp_dgain	手动 ISP 数字增益, 10bit 小数精度, 默认值为 0x400。 取值范围: [0x400, 0x40000], 具体范围与 sensor 相关。

【注意事项】

- 手动模式下, 手动曝光时间和增益的值会受到自动模式下最大/最小曝光时间和增益的限制。
- 手动曝光使能参数有效时, 必须设置相应的手动曝光参数, 若不设置, 则采用系统默认值。
- 增益单位为 10bit 小数精度的倍数, 即 1024 代表 1 倍。

- 若曝光参数设置超出最大（小）值，将使用 sensor 支持的最大（小）值代替。

【相关数据类型及接口】

[xmedia_ae_set_exposure_attr](#)

xmedia_ae_exposure_attr

【说明】

定义 ISP 曝光属性。

【定义】

```
typedef struct {
    xmedia_bool        bypass;
    xmedia\_video\_operation\_mode    op_type;
    xmedia_u8          ae_run_interval;
    xmedia_bool        hist_stat_adjust;
    xmedia_bool        ae_route_ex_valid;
    xmedia\_ae\_prior\_frame prior_frame;
    xmedia_bool        ae_gain_sep_cfg;
    xmedia\_ae\_manual\_attr manual_attr;
    xmedia\_ae\_auto\_attr auto_attr;
} xmedia_ae_exposure_attr;
```

【成员】

成员名称	描述
bypass	AE 模块 bypass 功能使能，默认为 XMEDIA_FALSE。
op_type	自动曝光或手动曝光开关， 默认为 XMEDIA_VIDEO_OPERATION_MODE_AUTO。
ae_run_interval	AE 算法运行的间隔，取值范围为[1, 255]。 取值为 1 时表示每帧都运行 AE 算法； 取值为 2 时表示每 2 帧运行 1 次 AE 算法，依此类推。 建议该值设置不要大于 2，否则 AE 调节速度会受到影响。 WDR 模式时，该值建议设置为 1，这样 AE 收敛会更加平滑。 该值默认为 1。
hist_stat_adjust	1024 段直方图统计方式是否根据场景进行调整开关，默认为 XMEDIA_TRUE。

成员名称	描述
ae_route_ex_valid	AE 扩展分配路线是否生效开关, XMEDIA_TRUE 时使用 AE 扩展分配路线, 否则使用普通 AE 分配路线。 默认为 XMEDIA_FALSE。
prior_frame	WDR 模式下曝光路线生效的优先帧, 线性模式下此参数需配置为 XMEDIA_LONG_FRAME, 当前 AE 库仅支持长帧优先模式。
ae_gain_sep_cfg	长短帧增益是否分开分配。
manual_attr	手动曝光属性结构体。
auto_attr	自动曝光属性结构体。

【注意事项】

- bypass 为 XMEDIA_TRUE 时, AE 模块被 bypass, 任何 AE 配置都不会对图像亮度产生影响。AE 计算结果保持为 AE bypass 前一帧的值。
- XMEDIA_AE 主要基于全局 1024 段直方图信息进行亮度调整, 正常场景 1024 段直方图只统计 Gb 或 Gr 分量。hist_stat_adjust 为 XMEDIA_TRUE 时, 若进入了大面积单色场景 (如大面积红色或蓝色), AE 算法会根据全局平均值信息对 1024 段直方图统计方式进行调整, 把 R 分量或 B 分量也考虑进来, 如此可以防止大面积红色或蓝色场景画面亮度偏高。hist_stat_adjust 为 XMEDIA_FALSE 时, AE 算法不会调整 1024 段直方图统计方式。红外场景建议将 hist_stat_adjust 设置为 XMEDIA_FALSE。
- WDR/线性模式切换时, ae_run_interval 会重置为 1, 用户可以在切换完成后, 根据需要修改该值。
- ae_gain_sep_cfg 仅在 2in1 WDR 模式下配置生效。
- 手动模式不支持长短帧增益分开分配。
- 曝光比变化过程中, 切换 ae_gain_sep_cfg 会交叉影响长短帧的曝光时间限制, 可能会导致图像亮度突变, 建议在曝光比收敛状态下切换 ae_gain_sep_cfg。

【相关数据类型及接口】

[xmedia_ae_set_exposure_attr](#)
[xmedia_ae_get_exposure_attr](#)

xmedia_ae_wdr_exposure_attr

【说明】

定义 WDR 模式下的曝光属性。

【定义】

```
typedef struct {
    xmedia_video_operation_mode exp_ratio_type;
    xmedia_u32 exp_ratio[XMEDIA_ISP_WDR_EXP_RATIO_MAX_NUM];
    xmedia_u32 exp_ratio_max;
    xmedia_u32 exp_ratio_min;
    xmedia_u16 tolerance;
    xmedia_u16 speed;
    xmedia_u16 ratio_bias;
} xmedia_ae_wdr_exposure_attr;
```



说明

宏定义如下：

```
#define XMEDIA_ISP_WDR_EXP_RATIO_MAX_NUM 3
```

【成员】

成员名称	描述
exp_ratio_type	仅在多帧合成 WDR 模式下有效。 XMEDIA_VIDEO_OPERATION_MODE_AUTO：根据场景自动计算长短帧曝光比； XMEDIA_VIDEO_OPERATION_MODE_MANUAL：手动配置长短帧曝光比。
exp_ratio	仅在多帧合成 WDR 模式下有效。 当 exp_ratio_type 为 XMEDIA_VIDEO_OPERATION_MODE_AUTO 时，exp_ratio 无效。 当 exp_ratio_type 为 XMEDIA_VIDEO_OPERATION_MODE_MANUAL 时，exp_ratio 为可读写，表示多帧合成 WDR 相邻 2 帧曝光比期望值。其中 exp_ratio[0]作为 S/VS 曝光比，exp_ratio[1]作为 M/S 曝光比，exp_ratio[2]作为 L/M 曝光比。上述 VS，S，M，L 分别表示极短帧，短帧，中帧和长帧的曝光时间，2 帧合成时只有 VS 和 S；3 帧合成时采用 VS，S 和 M；4 帧合成时采用 VS，S，M 和 L。6bit

成员名称	描述
	小数精度，0x40 表示曝光比为 1 倍。 取值范围：[0x40, 0xFFFF]
exp_ratio_max	仅在多帧合成 WDR 模式下有效。 当 exp_ratio_type 为 XMEDIA_VIDEO_OPERATION_MODE_AUTO 时，exp_ratio_max 表示最长帧与最短帧曝光时间比值的最大值。即 2 帧合成时表示 S/VS 曝光比的最大值，3 帧合成时表示 M/VS 曝光比的最大值，4 帧合成时表示 L/VS 曝光比的最大值。 当 exp_ratio_type 为 XMEDIA_VIDEO_OPERATION_MODE_MANUAL 时，exp_ratio_max 无效。 6bit 小数精度，0x40 表示曝光比为 1 倍。 取值范围：[0x40, 0x4000]
exp_ratio_min	仅在多帧合成 WDR 模式下有效。 当 exp_ratio_type 为 XMEDIA_VIDEO_OPERATION_MODE_AUTO 时，exp_ratio_min 表示长帧曝光时间与短帧曝光时间比值的最小值。 当 exp_ratio_type 为 XMEDIA_VIDEO_OPERATION_MODE_MANUAL 时，exp_ratio_min 无效。 格式为无符号 6.6bit 定点，0x40 表示长帧曝光时间与短帧曝光时间的比值为 1 倍。默认值为 0x40。 取值范围：[0x40, exp_ratio_max]
tolerance	曝光比容忍值，仅在两帧合成 WDR 模式下有效。 当 exp_ratio_type 为 XMEDIA_VIDEO_OPERATION_MODE_AUTO 时，该值越大，表示场景动态范围变化在一定范围内时，曝光比保持不变。默认值为 0xC。 取值范围：[0x0, 0xFF]
speed	自动曝光比调节速度，仅在两帧合成 WDR 模式下有效。 当 exp_ratio_type 为

成员名称	描述
	XMEDIA_VIDEO_OPERATION_MODE_AUTO 时，该值越大，自动曝光比调节速度越快。默认值为 0x20。 取值范围：[0x0, 0xFF]
ratio_bias	曝光比偏差值，仅在多帧合成 WDR 模式下有效。 当 exp_ratio_type 为 XMEDIA_VIDEO_OPERATION_MODE_AUTO 时，该值越大，自动曝光比越大。默认值为 0x400，表示不对自动曝光比算法的计算结果进行调整。经过该值调整的曝光比会受到曝光比最大/最小值的限制。 取值范围：[0x0, 0xFFFF]

【注意事项】

- 建议 tolerance 不要设置为太大的值，避免曝光比不能正确的反映场景动态范围变化。
- 针对某些对短帧最大曝光时间有限制的 sensor，曝光比较小时，长帧最大曝光时间较短，图像动态范围及噪声表现较差，导致自动曝光比计算不准确，此时建议限制最小曝光比，保证长帧最大曝光时间至少达到 3ms。如 Sony IMX274@30fps，短帧最大曝光时间为 680us，此时建议限制最小曝光比为 1:5，即设置 exp_ratio_min 为 0x140，以保证图像质量。
- 建议 speed 不要设置小于 0x8，避免某些场景由于计算精度不足导致曝光比调节太慢甚至不调。speed 过大可能导致曝光比变化太快，造成画面亮度出现振荡。
- 建议手动模式下 exp_ratio 不要设置大于 0x400。如果曝光比大于 0x400，在较亮的超宽动态场景，适当调高曝光比会优化长帧图像噪声表现。但是在较暗或低宽动态场景，如果曝光比过大，由于短帧最大曝光时间会被压缩，导致图像噪声表现会变差，而且会出现明显的噪声不连续，运动表现也会变差。
- **当前自动曝光比功能暂未开放，仅支持手动曝光比。**

【相关数据类型及接口】

[xmedia_ae_set_wdr_exposure_attr](#)
[xmedia_ae_get_wdr_exposure_attr](#)

xmedia_isp_ae_route_node

【说明】

定义 AE 分配路线节点属性。

【定义】

```
typedef struct {  
    xmedia_u32      int_time;  
    xmedia_u32      sys_gain;  
    xmedia_isp_iris_f_no iris_f_no;  
    xmedia_u32      iris_f_no_lin;  
} xmedia_isp_ae_route_node;
```

【成员】

成员名称	描述
int_time	节点曝光时间，单位为微秒(us)。 取值范围：(0x0, 0xFFFFFFFF]
sys_gain	节点增益，包括 sensor 模拟增益，sensor 数字增益和 ISP 数字增益，10bit 精度。 取值范围：[0x400, 0xFFFFFFFF]
iris_f_no	节点光圈 F 值大小，仅支持 P-Iris，不支持 DC-Iris。 取值范围：[XMEDIA_ISP_IRIS_F_NO_32_0, XMEDIA_ISP_IRIS_F_NO_1_0]。
iris_f_no_lin	节点光圈 F 值等效增益大小，仅支持 P-Iris，不支持 DC-Iris。 取值范围：[1, 1024]

【注意事项】

- 节点的曝光量是曝光时间、增益和光圈的乘积，节点曝光量必须单调递增，即后一个节点的曝光量大于或等于前一个节点的曝光量，第一个节点的曝光量最小，最后一个节点的曝光量最大。在计算曝光量时，光圈 F 值要等效成一个增益，公式如下：等效增益 $FNO = 1 \ll XMEDIA_ISP_IRIS_F_NO_XX_XX$ 。由此可知 F32.0 对应增益 1，F22.0 对应增益 2，F16.0 对应增益 4，以此类推，F1.0 对应增益 1024。
- 不支持设置等曝光量节点。

- 为了保证曝光节点的曝光量，曝光节点的一个分量发生限制时，会对其他未达到最大值的分量进行调整，实际生效路径可能与设定路径不一致。为了防止曝光量溢出，如果光圈分量使能，则同一节点的曝光时间与增益的乘积最大值不能超过 0x1FFFFFFFFFFFFFFF，如果光圈分量不使能，则同一节点的曝光时间与 10bit 精度系统增益的乘积最大值不能超过 0x7FFFFFFFFFFFFFFF。
- 如果相邻节点的曝光量增加，则应该有一个分量增加，其他分量固定，增加的分量决定该段路线的分配策略。例如增益分量增加，那么该段路线的分配策略是增益优先。
- 光圈分量仅支持 P-Iris，不支持 DC-Iris，因为 DC-Iris 无法精确控制。
- 针对 P-Iris，建议将第一个节点的曝光时间、增益和光圈 F 值都设置为相应的最小目标值，最后一个节点的曝光时间、增益和光圈 F 值都设置为相应的最大目标值。

【相关数据类型及接口】

[xmedia_ae_get_route_attr](#)

xmedia_isp_ae_route

【说明】

定义 AE 曝光分配策略属性。

【定义】

```
typedef struct {
    xmedia_u32          total_num;
    xmedia\_isp\_ae\_route\_node route_node[XMEDIA_ISP_AE_ROUTE_MAX_NODES];
} xmedia_isp_ae_route;
```



说明

宏定义如下：

```
#define XMEDIA_ISP_AE_ROUTE_MAX_NODES 16
```

【成员】

成员名称	描述
total_num	曝光分配路线节点数目，目前最大为 16。
route_node	曝光分配路线节点属性。

【注意事项】

- 最大支持 16 个节点，每个节点有曝光时间、增益、光圈三个分量，增益包含 sensor 模拟增益、sensor 数字增益、ISP 数字增益。
- 用户可以根据不同的场景设置不同的路线，分配路线支持动态切换。
- 针对 DC-Iris 和手动光圈镜头，默认 AE 分配策略是曝光时间优先，其次分配增益。如果当前曝光量不在用户设定的路线范围当中，按默认策略分配。在 DC_Iris 调节时，曝光量调节并不会参考 AE_ROUTE，因此实际生效的 AE_ROUTE 不会更新。
- 针对 P-Iris，默认 AE 分配策略是首先调节光圈，将光圈调至最大后调节曝光时间，最后再分配增益。如果当前曝光量不在用户设定的路线范围当中，按默认策略分配。
- 在线进行 DC-Iris 和 P-Iris 切换，AE route 会重置为与光圈类型相匹配的默认分配策略，用户可以根据需要在切换光圈类型时自行设置 AE route。默认分配策略的 AE route 参数与切换时的 AE 曝光时间最小/最大值，增益最大/最小值和光圈分量最大/最小值相对应。注意：由于 sensor.c 也可以设置 AE 分配路线，若要在线进行 DC-Iris 和 P-Iris 切换，请不要在 sensor.c 中给 iris_type 赋值，这样 AE 算法设置的光圈类型就会传递到 sensor.c 中，此时回调 sensor_get_ae_default 或 sensor_fps_set 等函数，才能得到与光圈类型相匹配的 AE route。

【相关数据类型及接口】

[xmedia_ae_set_route_attr](#)

[xmedia_ae_get_route_attr](#)

xmedia_ae_exp_info

【说明】

定义 ISP 曝光内部状态信息。

【定义】

```
typedef struct {
    xmedia_u32      exp_time;
    xmedia_u32      short_exp_time;
    xmedia_u32      median_exp_time;
    xmedia_u32      long_exp_time;
    xmedia_u32      again;
    xmedia_u32      dgain;
    xmedia_u32      again_sf;
    xmedia_u32      dgain_sf;
    xmedia_u32      isp_dgain;
    xmedia_u32      exposure;
```

```

xmedia_bool      exposure_is_max;
xmedia_s16       hist_error;
xmedia_u32       ae_hist[XMEDIA_ISP_AE_HIST_NUM];
xmedia_u8        avg_luma;
xmedia_u32       lines_per_500ms;
xmedia_u32       piris_f_no;
xmedia_u32       fps;
xmedia_u32       iso;
xmedia_u32       iso_sf;
xmedia_u32       iso_calibrate;
xmedia_u32       ref_exp_ratio;
xmedia_u32       first_stable_time;
xmedia_isp_ae_route ae_route;
xmedia_isp_ae_ext_route ae_route_ex;
xmedia_isp_ae_route ae_route_sf;
xmedia_isp_ae_ext_route ae_route_sf_ex;
} xmedia_ae_exp_info;

```



说明

宏定义如下：

```
#define XMEDIA_ISP_AE_HIST_NUM    1024
```

【成员】

成员名称	描述
exp_time	当前曝光时间，单位为微秒(us)。 取值范围：[0x0, 0xFFFFFFFF]
short_exp_time	WDR 模式下，表示当前短帧(S)曝光时间，单位为微秒(us)。线性模式不用关注该值。 取值范围：[0x0, 0xFFFFFFFF]
median_exp_time	WDR 模式下，表示当前中帧(M)曝光时间，单位为微秒(us)。线性模式不用关注该值。 取值范围：[0x0, 0xFFFFFFFF]
long_exp_time	WDR 模式下，表示当前长帧曝光时间，单位为微秒(us)。 取值范围：[0x0, 0xFFFFFFFF]
again	当前 sensor 模拟增益，10bit 小数精度。 取值范围：[0x400, 0xFFFFFFFF]

成员名称	描述
dgain	当前 sensor 数字增益, 10bit 小数精度。 取值范围: [0x400, 0xFFFFFFFF]
again_sf	当前 sensor 短帧模拟增益, 10bit 小数精度。 取值范围: [0x400, 0xFFFFFFFF]
dgain_sf	当前 sensor 短帧数字增益, 10bit 小数精度。 取值范围: [0x400, 0xFFFFFFFF]
isp_dgain	当前 ISP 数字增益, 10bit 小数精度。 取值范围: [0x400, 0xFFFFFFFF]
exposure	当前曝光量, 等于曝光时间与曝光增益的乘积, 其中 曝光时间的单位为曝光行数 , 增益精度 6bit。 取值范围: [0x0, 0xFFFFFFFF]
exposure_is_max	0: AE 未达到最大曝光水平; 1: AE 达到最大曝光水平。
hist_error	统计信息, AE 的目标亮度值与实际值的差, 该值为正表示当前期望的亮度信息大于实际的亮度信息, 该值为负表示期望的亮度信息小于实际的亮度信息。 取值范围: [-0x8000, 0x7FFF]
ae_hist	全局 1024 段直方图统计信息 取值范围: [0x0, 0xFFFFFFFF]
avg_luma	平均亮度信息 取值范围: [0x0, 0xFF]
lines_per_500ms	当前每 500ms 对应的曝光行数, 可用于将曝光时间的单位由 us 转换成行数。 取值范围: [0x0, 0xFFFFFFFF]
piris_f_no	当前 P-Iris 光圈 F 值对应的等效增益。 取值范围: [0x0, 0x400]
fps	实际图像帧率 * 100。

成员名称	描述
	取值范围：[0x0, 0xFFFFFFFF]
iso	当前 sensor 模拟增益*sensor 数字增益*ISP 数字增益*100，其中增益的精度都为 10bit。 取值范围：[0x64, 0xFFFFFFFF]
isof	当前 sensor 短帧模拟增益*sensor 短帧数字增益*ISP 短帧数字增益*100，其中增益的精度都为 10bit。 取值范围：[0x64, 0xFFFFFFFF]
iso_calibrate	标准 ISO，用于拍照 DCF 信息显示。 $\text{iso_calibrate} = \text{iso} * 256 / \text{u16ISOCalCoef.}$ 取值范围：[0x0, 0xFFFFFFFF]
ref_exp_ratio	参考曝光比，用于估计当前场景的动态范围，会受到 ISP_WDR_EXPOSURE_ATTR_S 中 Tolerance 和 Speed 等值的影响。 取值范围：[0x40, 0x4000]
first_stable_time	首次 AE 收敛稳定的时间，单位为微秒(us)， 暂未支持 取值范围：[0x0, 0xFFFFFFFF]
ae_route	实际生效的 AE route，各个节点中的曝光时间以 us 为单位，增益为 10bit 精度，光圈取值范围为 [ISP_IRIS_F_NO_32_0, ISP_IRIS_F_NO_1_0]。光圈类型为 DC-Iris 时，节点光圈值不会对曝光量分配产生影响。
ae_route_ex	实际生效的扩展 AE route，各个节点中的曝光时间以 us 为单位，增益为 10bit 精度，光圈取值范围为 [ISP_IRIS_F_NO_32_0, ISP_IRIS_F_NO_1_0]。光圈类型为 DC-Iris 时，节点光圈值不会对曝光量分配产生影响。
ae_route_sf	实际生效的短帧 AE route，各个节点中的曝光时间以 us 为单位，增益为 10bit 精度，光圈取值范围为 [ISP_IRIS_F_NO_32_0, ISP_IRIS_F_NO_1_0]。光圈类型为 DC-Iris 时，节点光圈值不会对曝光量分配产生影响。
ae_route_sf_ex	实际生效的短帧扩展 AE route，各个节点中的曝光时间以 us 为单位，增益为 10bit 精度，光圈取值范围为

成员名称	描述
	[ISP_IRIS_F_NO_32_0, ISP_IRIS_F_NO_1_0]。光圈类型为 DC-Iris 时，节点光圈值不会对曝光量分配产生影响。

【注意事项】

- 曝光量的计算未考虑光圈状态，为曝光时间、sensor 模拟增益、sensor 数字增益、ISP 数字增益 4 者乘积，其中曝光时间的单位为曝光行数，曝光增益为 6bit 小数精度。若该值的精度不足以满足需求，可以根据高精度的曝光时间(us)和增益(10bit 小数精度)重新计算一个曝光量。
- ISP 是否达到最大曝光水平的计算未考虑光圈状态。如果当前曝光时间大于等于最大目标曝光时间，当前增益大于等于最大目标增益，那么就认为 ISP 达到最大曝光水平，否则就未达到最大曝光水平。
- 图像平均亮度经过归一化处理，取值范围为 0 ~ 255。
- fps 可用于查询自动降帧状态下的实际帧率。
- again_sf、dgain_sf、isof 在线性模式下默认为 1024、1024、100，但从其他模式切换到线性模式时保留上一次的值不更新。
- 当长短帧增益分开配置时，iso、isof 的结果还包含对应帧的 WDRGain。
- WDR 模式下，exposure_is_max 表示长帧是否达到最大曝光量。

【相关数据类型及接口】

[xmedia_ae_query_exposure_info](#)

xmedia_isp_ae_ext_route_node

【说明】

定义 AE 扩展分配路线节点属性。

【定义】

```
typedef struct {
    xmedia_u32      int_time;
    xmedia_u32      again;
    xmedia_u32      dgain;
    xmedia_u32      isp_dgain;
    xmedia\_isp\_iris\_f\_no iris_f_no;
    xmedia_u32      iris_f_no_lin;
```

```
} xmedia_isp_ae_ext_route_node;
```

【成员】

成员名称	描述
int_time	节点曝光时间，单位为微秒(us)。 取值范围：(0x0, 0xFFFFFFFF]
again	sensor 模拟增益，10bit 精度。 取值范围：[0x400, 0xFFFFFFFF]
dgain	sensor 数字增益，10bit 精度。 取值范围：[0x400, 0xFFFFFFFF]
isp_dgain	ISP 数字增益，10bit 精度。 取值范围：[0x400, 0x40000]
iris_f_no	节点光圈 F 值大小，仅支持 P-Iris，不支持 DC-Iris。 取值范围：[ISP_IRIS_F_NO_32_0, ISP_IRIS_F_NO_1_0]
iris_f_no_lin	节点光圈 F 值等效增益大小，仅支持 P-Iris，不支持 DC-Iris。 取值范围：[1, 1024]

【注意事项】

- 节点的曝光量是曝光时间、sensor 模拟增益、sensor 数字增益、ISP 数字增益和光圈的乘积，节点曝光量必须单调递增，即后一个节点的曝光量大于或等于前一个节点的曝光量，第一个节点的曝光量最小，最后一个节点的曝光量最大。在计算曝光量时，光圈 F 值要等效成一个增益，公式如下：等效增益 $FNO = 1 \ll ISP_IRIS_F_NO_XX_XX$ 。由此可知 F32.0 对应增益 1，F22.0 对应增益 2，F16.0 对应增益 4，以此类推，F1.0 对应增益 1024。
- 不支持设置等曝光量节点。
- 为了保证曝光节点的曝光量，曝光节点的一个分量发生限制时，会对其他未达到最大值的分量进行调整，实际生效路径可能与设定路径不一致。
- 如果相邻节点的曝光量增加，则应该有一个分量增加，其他分量固定，增加的分量决定该段路线的分配策略。例如 sensor 模拟增益分量增加，那么该段路线的分配策略是 sensor 模拟增益优先。

- 光圈分量仅支持 P-Iris，不支持 DC-Iris，因为 DC-Iris 无法精确控制。
- 针对 P-Iris，建议将第一个节点的曝光时间、增益和光圈 F 值都设置为相应的最小目标值，最后一个节点的曝光时间、增益和光圈 F 值都设置为相应的最大目标值。
- 节点的曝光时间不能设置为 0，也不能设置过小，导致以 us 为单位的曝光时间对应的曝光行数为 0，否则可能产生异常。
- 节点增益的范围相比非扩展 AE route 有所缩小。此外，为防止曝光量溢出，sensor 模拟增益、sensor 数字增益和 ISP 数字增益的乘积等效为的 10bit 精度系统增益不能大于 0xFFFFFFFF。如果光圈分量使能，则同一节点的曝光时间与 10bit 精度系统增益的乘积最大值不能超过 0x1FFFFFFFFFFFFFFF，如果光圈分量不使能，则同一节点的曝光时间与 10bit 精度系统增益的乘积最大值不能超过 0x7FFFFFFFFFFFFFFF。
- 若某段路线需要设置 ISP 数字增益优先，请注意不要把 ISP 数字增益用满，因为 ISP 数字增益还有弥补分配精度的作用，建议至少留下 2 倍 ISP 数字增益的余量，供弥补精度使用，即分配路线中节点的 isp_dgain 不要大于 max_isp_dgain/2。
- 开启增益分开配置功能时，曝光节点中的 ISP 数字增益在实际分配时可能有部分被分配到 wdr_gain。

【相关数据类型及接口】

[xmedia_isp_ae_ext_route](#)

[xmedia_ae_set_ext_route_attr](#)

xmedia_isp_ae_ext_route

【说明】

定义 AE 曝光分配策略扩展属性。

【定义】

```
typedef struct {
    xmedia_u32                total_num;
    xmedia\_isp\_ae\_ext\_route\_node
    route_ex_node[XMEDIA_ISP_AE_EXT_ROUTE_MAX_NODES];
} xmedia_isp_ae_ext_route;
```



说明

宏定义如下：

```
#define XMEDIA_ISP_AE_EXT_ROUTE_MAX_NODES 16
```

【成员】

成员名称	描述
total_num	曝光扩展分配路线节点数目，目前最大为 16。
route_ex_node	曝光扩展分配路线节点属性。

【注意事项】

- 最大支持 16 个节点，每个节点有曝光时间、sensor 模拟增益、sensor 数字增益、ISP 数字增益和光圈五个分量。
- 用户可以根据不同的场景设置不同的路线，分配路线支持动态切换。

【相关数据类型及接口】

[xmedia_ae_set_ext_route_attr](#)

[xmedia_ae_get_ext_route_attr](#)

[xmedia_ae_smart_exposure_attr](#)

【说明】

定义智能模式下的曝光属性。

【定义】

```
typedef struct {
    xmedia_bool          enable;
    xmedia_bool          ir_mode;
    xmedia\_video\_operation\_mode smart_exp_type;
    xmedia_u8            iso_num;
    xmedia_u32           iso_level[XMEDIA\_ISP\_ISO\_MAX\_COUNT];
    xmedia_u8            luma_target[XMEDIA\_ISP\_ISO\_MAX\_COUNT];
    xmedia_u16           exp_coef[XMEDIA\_ISP\_ISO\_MAX\_COUNT];
    xmedia_u16           exp_coef_max;
    xmedia_u16           exp_coef_min;
    xmedia_u8            smart_interval;
    xmedia_u8            smart_speed;
    xmedia_u16           smart_delay_num;
} xmedia_ae_smart_exposure_attr;
```

【成员】

成员名称	描述
enable	智能 AE 是否使能，默认为 XMEDIA_FALSE。
ir_mode	是否为红外模式，默认为 XMEDIA_FALSE。 XMEDIA_TRUE 为红外模式； XMEDIA_FALSE 为普通模式。
smart_exp_type	XMEDIA_VIDEO_OPERATION_MODE_AUTO：根据目标检测结果自动调整曝光系数； XMEDIA_VIDEO_OPERATION_MODE_MANUAL：手动配置曝光系数。
iso_num	ISO 节点数量。
iso_level	ISO 节点等级。
luma_target	人脸目标亮度，此亮度为 YUV 图像中人脸区域 Y 分量均值。 取值范围：[0x0, 0xFF]
exp_coef	基于原始曝光调整的曝光系数，值越大，图像整体亮度越高； 值越小，图像整体亮度越低。 当 smart_exp_type 为 XMEDIA_VIDEO_OPERATION_MODE_AUTO 时，exp_coef 为 AE 算法根据目标检测结果计算的曝光系数。 当 exp_ratio_type 为 XMEDIA_VIDEO_OPERATION_MODE_MANUAL 时， exp_coef 为手动配置的曝光系数。 10bit 小数精度，0x400 表示曝光系数为 1 倍。 取值范围：[0x0, 0xFFFF]
exp_coef_max	智能曝光系数最大值。 - 当 smart_exp_type 为 XMEDIA_VIDEO_OPERATION_MODE_AUTO 时， exp_coef_max 表示智能曝光系数最大值。 - 当 smart_exp_type 为 XMEDIA_VIDEO_OPERATION_MODE_MANUAL 时， exp_coef_max 无效。

成员名称	描述
	10bit 小数精度，0x400 表示曝光系数最大值为 1 倍。 取值范围：[0x0, 0xFFFF]
exp_coef_min	智能曝光系数最小值。 当 smart_exp_type 为 XMEDIA_VIDEO_OPERATION_MODE_AUTO 时，exp_coef_min 表示智能曝光系数最小值。 当 smart_exp_type 为 XMEDIA_VIDEO_OPERATION_MODE_MANUAL 时，exp_coef_min 无效。 10bit 小数精度，0x400 表示曝光系数最小值为 1 倍。 取值范围：[0x0, 0xFFFF]
smart_interval	智能 AE 运行间隔。 取值为 1 时表示每帧运行，取值为 n 时，表示每 n 帧运行一次。 取值范围：[0x1, 0xFF]
smart_speed	智能 AE 调整速度。 默认值为 0x40，值越小，调整速度越慢；值越大，调整速度越快，也越容易出现震荡。 取值范围：[0x0, 0xFF]
smart_delay_num	智能 AE 延迟恢复帧数。 默认值为 0xB4，值越小，人离开场景后的曝光恢复时间越短，值越大，人离开场景后的曝光恢复时间越长。 取值范围：[0x0, 0x400]

【注意事项】

- 建议 luma_target 设置范围为[90,130]，如果设置的值过于不合理，会导致曝光调整结果与智能检测结果相互影响，出现图像亮度振荡。同样基于曝光调整结果与智能检测结果相互影响的考虑，收敛后人脸亮度可能在 luma_target 左右一定范围内，属正常现象。

- 建议 `exp_coef_max` 值不超过 0x1000,否则在人脸很暗的场景亮区会出现大面积过曝,而损失过多亮区细节。建议 `exp_coef_min` 值不超过 0x100,否则在人脸很亮的场景亮区会出现暗区过黑,而损失过多暗区细节。
- 智能 AE 运行间隔 `smart_interval` 受到 AE 自身运行间隔 `ae_run_interval` 影响,如 `smart_interval` 为 2, `ae_run_interval` 为 3,则智能 AE 间隔实际为两者乘积,每六帧运行一次。
- 智能 AE 运行间隔 `smart_interval` 建议与智能检测结果保持一致,如当前 Sensor 帧率为 25 帧,智能检测结果帧率为 8 帧,则建议运行间隔为 3。
- `xmedia_isp_set_smart_info` 接口调用需要和智能检测帧率保持一致,如智能检测结果帧率为 10 帧,则每秒调用 10 次 `xmedia_isp_set_smart_info`。
- 调整智能运行间隔 `smart_interval` 与智能调整速度 `smart_speed` 均可以调整收敛速度,当运行间隔较小,调整速度过大时,图像可能会出现亮度震荡,此时可适当降低运行速度。
- 智能 AE 延迟恢复帧数 `smart_delay_num` 可以调整人离开场景后的曝光恢复时间,该值设置过小时在人形人脸信息出现漏检的情况可能会导致图像亮度震荡,另外,适当设置较大值可以避免人连续进出的场景下图像亮度反复调节。
- 当人脸模型不可用,只有人形模型可用时,普通模式下计算的曝光系数不会低于 1 倍,红外模式下计算的曝光系数不会高于 1 倍。

【相关数据类型及接口】

`xmedia_ae_set_smart_exposure_attr`
`xmedia_ae_get_smart_exposure_attr`

6 AWB

6.1 概述

色温随可见光的光谱成分变化而变化，在低色温光源下，白色物体偏红，在高色温光源下，白色物体偏蓝。人眼可根据大脑的记忆判断，识别物体的真实颜色，AWB 算法的功能是降低外界光源对物体真实颜色的影响，使得我们采集的颜色信息转变为在理想日光光源下的无偏色信息。

6.2 重要概念

- 色温：色温是按绝对黑体来定义的，光源的辐射在可见区和绝对黑体的辐射完全相同时，此时黑体的温度就称此光源的色温。
- 白平衡：在不同色温的光源下，白色在传感器中的响应会偏蓝或偏红。白平衡算法通过调整 R, G, B 三个颜色通道的强度，使白色真实呈现。

6.3 功能描述

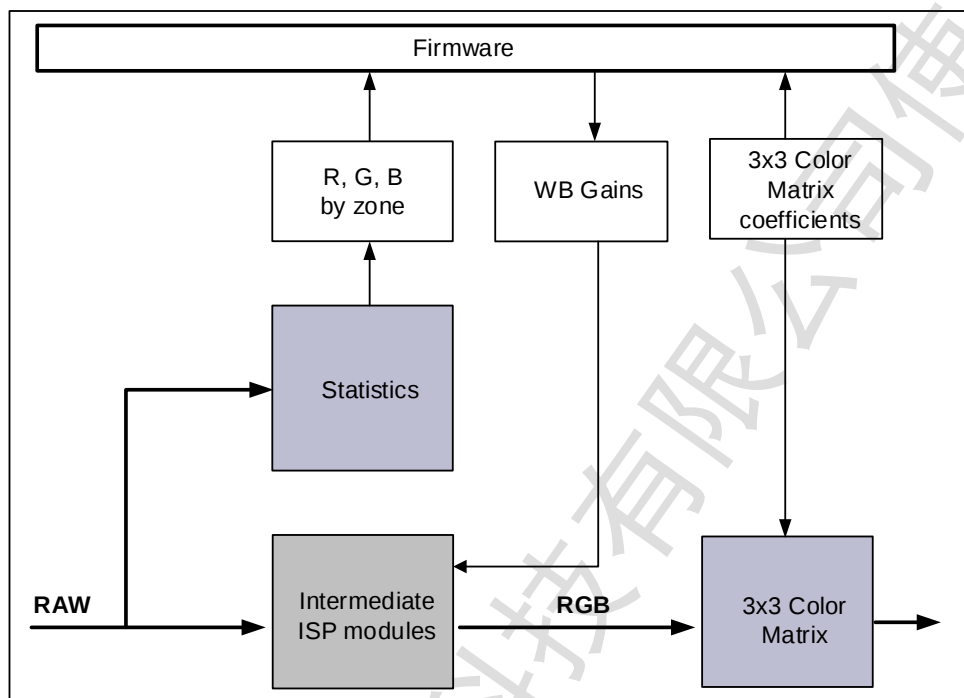
AWB 模块有硬件的 WB 信息统计模块及 AWB 策略控制算法 firmware 两部分组成。ISP 的 WB 信息统计模块判断 sensor 输出的每个像素是否满足用户设定的白点条件，计算所有满足条件的像素的 R、G、B 三个颜色通道平均值。

支持将图像分成 $M \times N$ (M 行 N 列) 区域，统计每个区域的 R、G、B 均值以及参与统计的白点个数。

支持输出整幅图像的 R、G、B 均值以及参与统计的白点个数。

AWB 工作原理如图 6-1 所示。

图6-1 AWB 工作原理图



6.4 API 参考

6.4.1 AWB 库接口

所有 AWB 库接口都只是针对 XMEDIA_AWB 库，如果客户自己实现 AWB 库，不需要关注这些接口，且无法使用这些接口。

- [xmedia_awb_register](#): 向 ISP 注册 AWB 库。
- [xmedia_awb_unregister](#): 向 ISP 反注册 AWB 库。

xmedia_awb_register

【描述】

向 ISP 注册 AWB 库。

【语法】

```
xmedia_s32 xmedia_awb_register(xmedia_u32 pipe);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_awb.h
- 库文件: libxmedia_isp.a、libxmedia_awb.a

【注意】

- 该接口调用了 ISP 库提供的 AWB 注册回调接口 xmedia_isp_register_awb_lib，以实现向 ISP 库注册的功能。
- 用户调用此接口完成 XMEDIA_AWB 库向 ISP 库注册。
- AWB 库可以注册多个实例，此接口不支持多进程操作。

【举例】

无

【相关主题】

[xmedia_awb_unregister](#)

xmedia_awb_unregister

【描述】

向 ISP 反注册 AWB 库。

【语法】

```
xmedia_s32 xmedia_awb_unregister(xmedia_u32 pipe);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_awb.h
- 库文件：libxmedia_isp.a、libxmedia_awb.a

【注意】

- 该接口调用了 ISP 库提供的 AWB 反注册回调接口。
xmedia_isp_unregister_awb_lib，以实现 AWB 向 ISP 库反注册的功能。
- 用户调用此接口完成 XMEDIA_AWB 库向 ISP 库反注册，此接口不支持多进程操作。

【举例】

无

【相关主题】

[xmedia_awb_register](#)

6.4.2 AWB 控制模块

- [xmedia_awb_set_attr](#)：设置白平衡属性。
- [xmedia_awb_get_attr](#)：获取白平衡属性。
- [xmedia_awb_set_scene_adj_attr](#)：设置 awb 场景调整属性参数。
- [xmedia_awb_get_scene_adj_attr](#)：获取 awb 场景调整属性参数。
- [xmedia_awb_set_ccm_attr](#)：设置 ccm 属性参数。
- [xmedia_awb_get_ccm_attr](#)：获取 ccm 属性参数。

- `xmedia_awb_set_saturation_attr`: 设置饱和度属性参数。
- `xmedia_awb_get_saturation_attr`: 获取饱和度属性参数。
- `xmedia_awb_query_info`: 更新 awb info 参数。
- `xmedia_awb_set_color_tone_attr`: 设置 color tone 属性参数。
- `xmedia_awb_get_color_tone_attr`: 获取 color tone 属性参数。
- `xmedia_awb_cal_gain_by_temp`: 根据色温值计算 awbgain。
- `xmedia_awb_set_color_sector_attr`: 设置颜色分区调节功能属性参数。
- `xmedia_awb_get_color_sector_attr`: 获取颜色分区调节功能属性参数。

`xmedia_awb_set_attr`

【描述】

设置白平衡属性。

【语法】

```
xmedia_s32 xmedia_awb_set_attr(xmedia_u32 pipe, xmedia_awb_attr *awb_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
awb_attr	白平衡的参数属性结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: `xmedia_awb.h`
- 库文件: `libxmedia_isp.a`、`libxmedia_awb.a`

【注意】

- 白平衡控制类型为自动时，AWB 算法自动调节白平衡系数。
- 白平衡控制类型为手动时，AWB 算法失效，需自行设定 rgain、ggain、bgain。
- 环境色温、照度会影响白点的分布范围，XMEDIA_AWB 库算法在运行过程中，会根据环境参数自动刷新 AWB Bayer 域统计信息的白点参数。用户希望在 XMEDIA_AWB 库算法运行时修改 AWB 统计参数，需要调用 xmedia_awb_set_attr 接口，关闭统计参数自动刷新功能。用户关闭统计参数自动刷新功能，到 ISP 收到配置并响应，有 2 帧的延时。因此，用户设置后需要等待 2 帧时间再修改 AWB 统计参数。

【举例】

无

【相关主题】

[xmedia_awb_get_attr](#)

xmedia_awb_get_attr

【描述】

获取白平衡属性。

【语法】

```
xmedia_s32 xmedia_awb_get_attr(xmedia_u32 pipe, xmedia_awb_attr *awb_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
awb_attr	白平衡的参数属性结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_awb.h
- 库文件: libxmedia_esp.a、libxmedia_awb.h

【注意】

无

【相关主题】

[xmedia_awb_set_attr](#)

xmedia_awb_set_scene_adj_attr

【描述】

设置 awb 场景调整属性。

【语法】

```
xmedia_s32 xmedia_awb_set_scene_adj_attr(xmedia_u32 pipe, const  
xmedia\_awb\_scene\_adj\_attr *awb_scene_adj_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
awb_scene_adj_attr	awb 场景调整属性结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_awb.h

- 库文件: libxmedia_isp.a、libxmedia_awb.h

【注意】

无

【相关主题】

[xmedia_awb_get_scene_adj_attr](#)

xmedia_awb_get_scene_adj_attr

【描述】

获取 awb 场景调整属性。

【语法】

```
xmedia_s32 xmedia_awb_get_scene_adj_attr(xmedia_u32 pipe,  
xmedia\_awb\_scene\_adj\_attr *awb_scene_adj_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
awb_scene_adj_attr	awb 场景调整属性结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_awb.h
- 库文件: libxmedia_isp.a、libxmedia_awb.h

【注意】

无

【举例】

无

【相关主题】

[xmedia_awb_set_scene_adj_attr](#)

`xmedia_awb_set_ccm_attr`

【描述】

设置颜色矩阵。

【语法】

```
xmedia_s32 xmedia_awb_set_ccm_attr(xmedia_u32 pipe, const xmedia\_awb\_ccm\_attr *ccm_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号	输入
ccm_attr	颜色矩阵结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_awb.h
- 库文件：libxmedia_isp.a

【注意】

- 颜色校正矩阵的数据格式，应与校正工具提供的保持一致。
- 可根据当前色温，设置不同的 CCM，从而在高低色温下都达到较好的颜色还原。

- 该 MPI 支持最大七个最小三个不同色温下的色彩还原矩阵，要按照色温递减的方式配置 CCM，即 0-6 组 CCM 矩阵由高色温到低色温。典型的三组 CCM 为 D50，TL84，A 三个光源下的 CCM。典型的五组 CCM 为 10K，D65，D50，TL84，A 五个光源下的 CCM。前一组的色温值与后一组的色温值保持降序排列。
- 手动模式下，ISP 系统生效的 CCM 即用户手动配置的 CCM，有利于 CCM 精调，因此建议在手动模式下校正不同色温光源下的 CCM 矩阵，得到多组 CCM 矩阵后再通过 MPI 写入，验证多个光源下的 CCM 组和效果。手动 CCM 模式下，饱和度调整有效。在特定光源下做 CCM 精调时，需要先确认饱和度是否期望值。

【举例】

无

【相关主题】

[xmedia_awb_get_ccm_attr](#)

xmedia_awb_get_ccm_attr

【描述】

获取颜色矩阵。

【语法】

```
xmedia_s32 xmedia_awb_get_ccm_attr(xmedia_u32 pipe, xmedia\_awb\_ccm\_attr *ccm_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号	输入
ccm_attr	颜色矩阵结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_awb.h
- 库文件: libxmedia_isp.a

【注意】

无

【举例】

无

【相关主题】

[xmedia_awb_set_ccm_attr](#)

xmedia_awb_set_saturation_attr

【描述】

设置颜色饱和度属性。

【语法】

```
xmedia_s32 xmedia_awb_set_saturation_attr(xmedia_u32 pipe, const xmedia_awb_saturation_attr
*sat_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
sat_attr	颜色饱和度属性结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_awb.h
- 库文件: libxmedia_isp.a、libxmedia_awb.a

【注意】

无

【举例】

无

【相关主题】

[xmedia_awb_get_saturation_attr](#)

xmedia_awb_get_saturation_attr

【描述】

获取颜色饱和度属性。

【语法】

```
xmedia_s32 xmedia_awb_get_saturation_attr(xmedia_u32 pipe, xmedia_awb_saturation_attr
*sat_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
sat_attr	颜色饱和度属性结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_awb.h

- 库文件: libxmedia_esp.a、libxmedia_awb.a

【注意】

无

【相关主题】

[xmedia_awb_set_saturation_attr](#)

xmedia_awb_set_color_tone_attr

【描述】

设置 RGB 通道色调强度的参数。

【语法】

```
xmedia_s32 xmedia_awb_set_color_tone_attr(xmedia_u32 pipe, const
xmedia_awb_color_tone_attr *ct_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ct_attr	设置 RGB 通道色调的参数。Rgain、Ggain、Bgain 分别增强三个通道的颜色强度。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_awb.h
- 库文件: libxmedia_esp.a、libxmedia_awb.a

【注意】

无

【相关主题】

[xmedia_awb_get_color_tone_attr](#)

xmedia_awb_get_color_tone_attr

【描述】

获取 RGB 通道色调的参数。

【语法】

```
xmedia_s32 xmedia_awb_set_color_tone_attr(xmedia_u32 pipe,  
xmedia\_awb\_color\_tone\_attr *ct_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ct_attr	获取 RGB 通道色调的参数。Rgain、Ggain、Bgain 分别增强三个通道的颜色强度。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_awb.h
- 库文件: libxmedia_isp.a、libxmedia_awb.a

【注意】

无

【相关主题】

[xmedia_awb_set_color_tone_attr](#)

xmedia_awb_query_info

【描述】

获取当前白平衡增益系数，检测色温，饱和度值，颜色校正矩阵系数。

【语法】

```
xmedia_s32 xmedia_awb_query_info(xmedia_u32 pipe, xmedia_awb_info *wb_info);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
wb_info	颜色相关状态参数结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_awb.h
- 库文件：libxmedia_isp.a

【注意】

若用户使用非 XMEDIA_AWB 库算法，则该接口需要自己实现并且需要同步修改 PQTOOLS 的 xml 文件，具体修改方式请参考《图像质量调试工具使用指南》

【举例】

无

【相关主题】

无

xmedia_awb_cal_gain_by_temp

【描述】

计算特定色温下的白平衡增益系数。

【语法】

```
xmedia_s32 xmedia_awb_cal_gain_by_temp(xmedia_u32 pipe, const xmedia_awb_attr *awb_attr,
xmedia_u16 color_temp, xmedia_s16 shift, xmedia_u16 *awb_gain);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
awb_attr	白平衡的参数属性。需要用到 xmedia_awb_attr 中的 static_wb, curve_para, gain_norm_en 三组参数。	输入
color_temp	色温值，单位为 Kelvin。 取值范围：[0, 65535]	输入
shift	白点与 Planckian 曲线的位置和距离。 取值范围：[-64, 64]。	输入
awb_gain	预设色温下的 R, Gr, Gb, B 四个通道增益。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_awb.h
- 库文件：libxmedia_isp.a, libxmedia_awb.a

【注意】

- 白平衡标定参数决定预设色温增益计算的准确性，因此，在调用该接口前，必须完成标定，且标定参数配置到 ISP。用户先通过 xmedia_awb_set_attr 接口获取标定结果，再调用 xmedia_awb_cal_gain_by_temp 接口计算预设色温，过程中不能再修改标定结果。
- xmedia_awb_cal_gain_by_temp 功能基于 AWB 标定参数实现，因此，该接口仅对结构体的标定参数 static_wb、curve_para 做有效范围检查，不对其他参数做约束。
- xmedia_awb_attr 结构体的 gain_norm 参数会影响白平衡增益值。gain_norm 为 0 时，最小增益为 0x100；gain_norm 为 1 时，最小增益和 sensor 黑电平相关。
- shift 参数决定光源点与 Planckian 曲线的位置关系，shift 为负数时，光源点位于 Planckian 曲线左侧，计算的白平衡增益在预设色温下会稍微偏红；shift 为正数时，光源点位于 Planckian 曲线右侧，计算的白平衡增益在预设色温下会稍微偏绿。

【举例】

无

【相关主题】

[xmedia_awb_set_attr](#)

xmedia_awb_set_color_sector_attr

【描述】

设置颜色分区调节功能属性参数。

【语法】

```
xmedia_s32 xmedia_awb_set_color_sector_attr(xmedia_u32 pipe, const xmedia_color_sector_attr * color_sector_attr)
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
color_sector_attr	颜色分区调节属性参数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_awb.h
- 库文件: libxmedia_esp.a、libxmedia_awb.a

【注意】

XM7606V13 调试无效。

【举例】

无

【相关主题】

无

xmedia_awb_get_color_sector_attr

【描述】

获取颜色分区调节功能属性参数。

【语法】

```
xmedia_s32 xmedia_awb_get_color_sector_attr(xmedia_u32 pipe, xmedia_color_sector_attr *
color_sector_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
color_sector_attr	颜色分区调节属性参数。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_awb.h
- 库文件: libxmedia_isp.a、libxmedia_awb.h

【注意】

XM7606V13 暂未支持。

【相关主题】

无

6.5 数据类型

6.5.1 Register

- [xmedia_sensor_register_awb_info](#): 定义 sensor 驱动与 AWB 库之间接口结构体。
- [xmedia_sensor_register_awb_func](#): 定义 sensor 回调函数结构体。
- [xmedia_isp_awb_sensor_default](#): 定义 AWB 算法库的初始化参数结构体。
- [xmedia_isp_awb_agc_table](#): 定义 AWB 初始化动态增益控制饱和度数组。
- [xmedia_isp_awb_ccm](#): 定义 AWB 初始化 ccm 矩阵配置。
- [xmedia_isp_awb_ccm_table](#): 定义 AWB 初始化不同色温下 ccm 矩阵。

xmedia_sensor_register_awb_info

【说明】

定义 sensor 驱动与 AWB 库之间接口结构体。

【定义】

```
typedef struct {
    xmedia_u32                dev_id;
    xmedia_sensor_register_awb_func  func;
}
```

```
} xmedia_sensor_register_awb_info;
```

【成员】

成员名称	描述
dev_id	sensor 设备号。
func	sensor 库提供的回调函数结构体。

【差异说明】

无。

【注意事项】

无。

【相关数据类型及接口】

[xmedia_sensor_register_awb_func](#)

xmedia_sensor_register_awb_func

【说明】

定义 sensor 回调函数结构体。

【定义】

```
typedef struct {  
    xmedia_s32 (*pfn_sensor_get_awb_default)(xmedia_u32 dev,  
    xmedia_isp_awb_sensor_default *awb_sns_dft);  
} xmedia_sensor_register_awb_func;
```

【成员】

成员名称	描述
pfn_sensor_get_awb_default	获取 awb 默认参数的回调函数指针。

【注意事项】

如果回调函数指针不需要赋值，需要置为 NULL。

【相关数据类型及接口】

[xmedia_isp_awb_sensor_default](#)

xmedia_isp_awb_sensor_default

【说明】

定义 AWB 算法库的初始化参数结构体。

【定义】

```
typedef struct {  
    xmedia_u16          wb_ref_temp;  
    xmedia_u16          gain_offset[XMEDIA_ISP_BAYER_PATTERN_NUM];  
    xmedia_s32          wb_para[XMEDIA_ISP_AWB_CURVE_PARA_NUM];  
    xmedia_u16          golden_rgain;  
    xmedia_u16          golden_bgain;  
    xmedia_u16          sample_rgain;  
    xmedia_u16          sample_bgain;  
    xmedia_isp_awb_agc_table agc_tbl;  
    xmedia_isp_awb_ccm      ccm;  
    xmedia_u16          init_rgain;  
    xmedia_u16          init_ggain;  
    xmedia_u16          init_bgain;  
    xmedia_u8           awb_runinterval;  
    xmedia_u16          init_ccm[XMEDIA_ISP_CCM_MATRIX_SIZE];  
    xmedia_video_operation_mode wb_op_type;  
    xmedia_video_operation_mode ccm_op_type;  
    xmedia_video_operation_mode sat_op_type;  
} xmedia_isp_awb_sensor_default;
```

【成员】

成员名称	描述
wb_ref_temp	校正静态白平衡系数的光源色温值。 推荐选用 5000K 附近光源进行静态白平衡校正。
gain_offset	静态白平衡系数。 取值范围：[0x0, 0xFFF]
wb_para	校准曲线的系数。 取值范围：curve_para[0-5] != 0; curve_para[4] = 128。
golden_rgain	Golden 样机 AWB 在线标定得到的 G/R 值。
golden_bgain	Golden 样机 AWB 在线标定得到的 G/B 值。

成员名称	描述
sample_rgain	当前样机 AWB 在线标定得到的 G/R 值。
sample_bgain	当前样机 AWB 在线标定得到的 G/B 值。
agc_tbl	根据增益动态调节饱和度的插值数组，取值范围为 [0,255]。
ccm	不同色温下的颜色校正矩阵和对应的色温值。
init_rgain	ISP 启动时 AWB 初始 R 通道增益。
init_ggain	ISP 启动时 AWB 初始 G 通道增益。
init_bgain	ISP 启动时 AWB 初始 B 通道增益。
awb_runinterval	ISP 启动时 AWB 工作频率。取值范围： [0x1, 0xFF]
init_ccm	ISP 启动时 CCM 初始值。
wb_op_type	ISP 启动时 wb 自动/手动模式选择。
ccm_op_type	ISP 启动时 ccm 自动/手动模式选择。
sat_op_type	ISP 启动时 saturation 自动/手动模式选择。

【注意事项】

无。

【相关数据类型及接口】

[xmedia_isp_awb_agc_table](#)

xmedia_isp_awb_agc_table

【说明】

定义 AWB 初始化动态增益控制饱和度数组。

【定义】

```
typedef struct {
    xmedia_bool    valid;
    xmedia_u8      saturation[XMEDIA_ISP_AUTO_ISO_STRENGTH_NUM];
} xmedia_isp_awb_agc_table;
```

【成员】

成员名称	描述
valid	该结构体的数据是否有效，取值范围为[0,1]。
saturation	根据增益动态调节饱和度的插值数组，取值范围为[0, 255]。

【注意事项】

无。

【相关数据类型及接口】

无。

xmedia_isp_awb_ccm

【说明】

定义 AWB 初始化 ccm 矩阵配置。

【定义】

```
typedef struct {
    xmedia_u16          ccm_tab_num;
    xmedia_isp_awb_ccm_table ccm_tab[XMEDIA_ISP_CCM_TAB_MAX_NUM];
} xmedia_isp_awb_ccm;
```

【成员】

成员名称	描述
ccm_tab_num	当前配置的 CCM 的组数。取值范围： [3, 7]。
ccm_tab	不同色温下的颜色校正矩阵和对应的 ccm table。

【注意事项】

无。

【相关数据类型及接口】

[xmedia_isp_awb_ccm_table](#)

xmedia_isp_awb_ccm_table

【说明】

定义 AWB 初始化不同色温下 ccm 矩阵。

【定义】

```
typedef struct {  
    xmedia_u16 ct;  
    xmedia_u16 ccm[XMEDIA_ISP_CCM_MATRIX_SIZE];  
} xmedia_isp_awb_ccm_table;
```

【成员】

成员名称	描述
ct	不同色温下的颜色校正矩阵和对应的色温。
ccm	不同色温下的颜色校正矩阵和对应的 ccm 矩阵。

【注意事项】

无。

【相关数据类型及接口】

无。

6.5.2 WB

- [xmedia_awb_alg](#): 定义白平衡的计算方式属性。
- [xmedia_awb_manual_attr](#): 定义 ISP 手动白平衡属性。
- [xmedia_awb_normal_alg_type](#): 定义白平衡算法类型。
- [xmedia_awb_ct_limit_attr](#): 定义白平衡的增益范围限制属性。
- [xmedia_awb_cbcr_track_attr](#): 定义 Bayer 域统计信息的联动参数。
- [xmedia_awb_luma_hist_attr](#): 定义白平衡的亮度直方图统计参数。
- [xmedia_awb_auto_attr](#): 定义 ISP 自动白平衡属性。
- [xmedia_awb_attr](#): 定义白平衡属性。
- [xmedia_awb_extra_lightsource_info](#): 定义独立光源点的信息。
- [xmedia_awb_indoor_outdoor_status](#): 定义 AWB 室内外状态。
- [xmedia_awb_in_out_attr](#): 定义对场景做室内外判断的参数。
- [xmedia_awb_multi_ls_type](#): 定义混合光源下的 AWB 策略。
- [xmedia_awb_scene_adj_attr](#): 定义 awb 场景调整属性。

- `xmedia_awb_ccm_manual`: 定义手动颜色校正矩阵属性。
- `xmedia_awb_ccm_param`: 定义多组颜色矫正矩阵系数
- `xmedia_awb_ccm_auto`: 定义自动颜色校正矩阵属性。
- `xmedia_awb_ccm_attr`: 定义 ISP 颜色矩阵属性。
- `xmedia_awb_saturation_manual`: 定义手动饱和度属性。
- `xmedia_awb_saturation_auto`: 定义自动饱和度属性。
- `xmedia_awb_saturation_attr`: 定义 ISP 颜色饱和度属性。
- `xmedia_awb_info`: 定义白平衡, 饱和度, 颜色校正信息。
- `xmedia_awb_color_tone_attr`: colortone 提供色调调节接口。
- `xmedia_color_sector_manual`: 定义手动颜色分区调节功能调整属性。
- `xmedia_color_sector_auto`: 定义颜色分区调节功能调整属性。
- `xmedia_color_sector_attr`: 定义颜色分区调节功能调整属性。

xmedia_awb_alg

【说明】

awb 算法类型选择, 可选传统模式和 ai-awb 模式。

【定义】

```
typedef enum {
    XMEDIA_AWB_ALG_NORMAL = 0,
    XMEDIA_AWB_ALG_AI      = 1,
    XMEDIA_AWB_ALG_MAX
} xmedia_awb_alg;
```

【成员】

成员名称	描述
<code>XMEDIA_AWB_ALG_NORMAL</code>	传统 awb 算法模式
<code>XMEDIA_AWB_ALG_AI</code>	ai-awb 模式

【差异说明】

暂未支持 `XMEDIA_AWB_ALG_AI`。

【注意事项】

无

【相关数据类型及接口】

[xmedia_awb_attr](#)

xmedia_awb_manual_attr

【说明】

定义 ISP 手动白平衡属性。

【定义】

```
typedef struct {  
    xmedia_u16 rgain;  
    xmedia_u16 grgain;  
    xmedia_u16 gbgain;  
    xmedia_u16 bgain;  
} xmedia_awb_manual_attr;
```

【成员】

成员名称	描述
rgain	手动白平衡 R 通道增益，8bit 小数精度。 取值范围：[0x0,0xFFF]
rggain	手动白平衡 Gr 通道增益，8bit 小数精度。 取值范围：[0x0,0xFFF]
gbgain	手动白平衡 Gb 通道增益，8bit 小数精度。 取值范围：[0x0,0xFFF]
bgain	手动白平衡 B 通道增益，8bit 小数精度。 取值范围：[0x0,0xFFF]

【注意事项】

无

【相关数据类型及接口】

[xmedia_awb_attr](#)

xmedia_awb_normal_alg_type

【说明】

定义白平衡的算法类型属性。

【定义】

```
typedef enum {
    XMEDIA_AWB_NORMAL_ALG_LOWCOST = 0,
    XMEDIA_AWB_NORMAL_ALG_ADVANCE = 1,
    XMEDIA_AWB_NORMAL_ALG_MAX
} xmedia_awb_normal_alg_type;
```

【成员】

成员名称	描述
XMEDIA_AWB_NORMAL_ALG_LOWCOST	低成本灰度世界算法，根据统计信息自动计算区间权重的 AWB 算法。
XMEDIA_AWB_NORMAL_ALG_ADVANCE	对统计信息分类，再次筛选白色块的 AWB 算法。

【注意事项】

- XMEDIA_AWB_NORMAL_ALG_LOWCOST CPU 占用低：AWB 不同照度的稳定性较好，对光源的适应性稍好。
- XMEDIA_AWB_NORMAL_ALG_ADVANCE CPU 占用高：高低色温的极低照度情况下，保留轻微光源色；提高了大面积纯色场景（室外大面积绿色、室内大面积肤色等）AWB 算法准确度。

【相关数据类型及接口】

[xmedia_awb_auto_attr](#)

xmedia_awb_ct_limit_attr

【说明】

定义白平衡的增益范围限制属性。

【定义】

```
typedef struct {
```

```

xmedia_bool      enable;
xmedia_video_operation_mode op_type;
xmedia_u16 high_rg_limit;
xmedia_u16 high_bg_limit;
xmedia_u16 low_rg_limit;
xmedia_u16 low_bg_limit;
} xmedia_awb_ct_limit_attr;

```

【成员】

成员名称	描述
enable	白平衡的增益范围限制开关。
op_type	自动或手动设定白平衡的增益限制。
high_rg_limit	手动模式下用户设定高色温下的最大 R 增益，8bit 小数精度，取值范围：[0x0, 0xFFFF]
high_bg_limit	手动模式下用户设定高色温下的最小 B 增益，8bit 小数精度，取值范围：[0x0, 0xFFFF]
low_rg_limit	手动模式下用户设定低色温下的最小 R 增益，8bit 小数精度，取值范围：[0x0, high_rg_limit)
low_bg_limit	手动模式下用户设定低色温下的最大 B 增益，8bit 小数精度，取值范围：(high_bg_limit, 0xFFFF]

【注意事项】

- xmedia_awb_ct_limit_attr 结构体定义环境色温超过用户设定的色温范围时，AWB 应采取的动作。AWB 检测到环境色温超过用户设置的上下限时，相关参数才会生效。
- 支持自动模式和手动模式选择。自动模式下，AWB 根据用户校正的色温曲线计算色温上下限色温的 AWB 增益，以限制 R, B 通道增益；手动模式下，高色温时，AWB 调用 high_rg_limit、high_bg_limit 参数限制 R, B 通道增益，低色温时，AWB 调用 low_rg_limit、low_bg_limit 参数限制 R, B 通道增益。
- 建议用户在进行 AWB 色温曲线校正时，确定了 AWB 工作的色温范围后，再确定 high_rg_limit、high_bg_limit、low_rg_limit、low_bg_limit 四个参数取值。Rg 对应 Planckian 曲线的横坐标值，Bg 对应 Planckian 曲线的纵坐标值。

【相关数据类型及接口】

xmedia_awb_auto_attr

xmedia_awb_cbcr_track_attr

【说明】

定义 Bayer 域统计信息的联动参数。

【定义】

```
typedef struct {  
    xmedia_bool enable;  
    xmedia_u8 iso_num;  
    xmedia_u32 iso_level[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u16 crmax[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u16 crmin[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u16 cbmax[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u16 cbmin[XMEDIA_ISP_ISO_MAX_COUNT];  
} xmedia_awb_cbcr_track_attr;
```

【成员】

成员名称	描述
enable	Bayer 域统计信息参数与环境照度、色温的联动使能开关。
iso_num	ISO 数量。
iso_level	ISO 等级。
crmax	不同照度下的 CrMax 取值，取值范围：[0x0, 0xFFFF]。
crmin	不同照度下的 CrMin 取值，取值范围：[0x0, crmax]。
cbmax	不同照度下的 CbMax 取值，取值范围：[0x0, 0xFFFF]。
cbmin	不同照度下的 CbMin 取值，取值范围：[0x0, cbmax]。

【注意事项】

- 统计参数联动使能后，AWB 算法会根据环境照度、色温、用户设置的 crmax 数组等实时计算 crmax、crmin、cbmax、cbmin 四个参数，并配置相应的逻辑寄存器，此时用户通过 PQ Tools 设置以上四个统计参数不生效。

- 统计参数联动使能，AWB 算法计算的 CrMax 等统计参数在根据 ISO 插值的基础上还和环境色温相关。低色温时，白点范围较宽；中高色温时，白点范围较窄。
- 用户手动配置 crmax、crmin、cbmax、cbmin 四个参数时，需要先关闭联动功能。
- crmax[0]、crmin[0]、cbmax[0]、cbmin[0]的取值可在 AWB 参数标定时确定。用户确定了支持的色温范围后，捕获高低色温的 RAW 图片，计算白色区域的 R/G、B/G 值。crmax[0]和 cbmin[0]分别对应低色温的 R/G、B/G；crmin[0]和 cbmax[0]分别对应高色温的 R/G、B/G。推荐用户设置的 Cr、Cb 范围稍大于 RAW 图片统计的 R/G、B/G 取值范围。
- AWB 算法计算的 crmax 等统计参数在根据 ISO 插值的时候，会基于 crmax[0]、crmin[0]、cbmax[0]、cbmin[0]的值。所以不同 ISO 下的取值设定应以 ISO100 的值为基础，单调递增或递减。
- 建议在低色温(钠灯)环境标定 crmax、cbmin 数组。用户统计不同照度下白色区域的 R/G、B/G 值，设置 crmax、cbmin 数组。推荐用户设置的 Cr、Cb 范围稍大于 RAW 图片统计的 R/G、B/G 取值范围。
- 因低照度下环境色温多在 5000K 以下，crmin、cbmax 两个数组的取值可设为常数。
- 当 AWB gain 配置在 Sensor 处；AWB gain 配置在 WDR 合成前的 DG1 处；AWB 统计信息配置在 DRC 后的时候，在这三种情况下，xmedia_awb_cbcr_track_attr 参数不生效，Min 固定为 128，Max 固定为 512。

【相关数据类型及接口】

[xmedia_awb_auto_attr](#)

[xmedia_awb_attr](#)

[xmedia_awb_luma_hist_attr](#)

【说明】

定义白平衡的亮度直方图统计参数。

【定义】

```
typedef struct {
    xmedia_bool          enable;
    xmedia_video_operation_mode op_type;
    xmedia_u8            hist_thresh[XMEDIA_AWB_LUM_HIST_NUM];
    xmedia_u16           hist_wt[XMEDIA_AWB_LUM_HIST_NUM];
} xmedia_awb_luma_hist_attr;
```



说明

宏定义如下：

```
#define XMEDIA_AWB_LUM_HIST_NUM
```

6

【成员】

成员名称	描述
enable	亮度是否影响分块的权重。
op_type	自动模式下，AWB 算法对分块统计结果做亮度直方图统计，自动分配亮度权重。手动模式下，用户设置亮度直方图的门限和权重。
hist_thresh	用户设置亮度直方图的门限，仅手动模式有效。
hist_wt	用户设置亮度直方图的权重，自动模式和手动模式下均有效。8bit 小数精度。

【注意事项】

- hist_thresh [0]固定为 0，hist_thresh [5]固定为 0xFF。hist_thresh [i+1]取值应大于 hist_thresh [i]。
- hist_wt 参数用来设置 hist_thresh 的权重。对应用户可通过 hist_wt 设置实现亮区优先、暗区优先。

【相关数据类型及接口】

[xmedia_awb_auto_attr](#)

xmedia_awb_auto_attr

【说明】

定义白平衡自动属性。

【定义】

```
typedef struct {  
    xmedia_bool        enable;  
    xmedia_u16          ref_ct;  
    xmedia_u16          static_wb[XMEDIA_AWB_BAYER_CHN_NUM];  
    xmedia_s32          curve_para[XMEDIA_ISP_AWB_CURVE_PARA_NUM];  
    xmedia\_awb\_normal\_alg\_type alg_type;  
    xmedia_u8           rg_str;  
    xmedia_u8           bg_str;  
    xmedia_u16          speed;  
    xmedia_u16          zone_sel;  
}
```

```

xmedia_u16          high_ct;
xmedia_u16          low_ct;
xmedia_awb_ct_limit_attr  ct_limit;
xmedia_bool         shift_limit_en;
xmedia_u8           shift_limit;
xmedia_bool         gain_norm_en;
xmedia_bool         natural_cast_en;
xmedia_awb_cbcr_track_attr cbcr_track;
xmedia_awb_luma_hist_attr luma_hist;
xmedia_bool         awb_zone_wt_en;
xmedia_u8           zone_wt[XMEDIA_AWB_ZONE_ORIG_ROW *
XMEDIA_AWB_ZONE_ORIG_COLUMN];
} xmedia_awb_auto_attr;

```



说明

宏定义如下：

- #define XMEDIA_AWB_BAYER_CHN_NUM 4
- #define XMEDIA_ISP_AWB_CURVE_PARA_NUM 6
- #define XMEDIA_AWB_ZONE_ORIG_ROW 32
- #define XMEDIA_AWB_ZONE_ORIG_COLUMN 32

【成员】

成员名称	描述
enable	自动白平衡使能，默认 XMEDIA_TRUE。
ref_ct	校正静态白平衡系数的光源色温值。 推荐选用 5000K 附近光源进行静态白平衡校正。
static_wb	静态白平衡系数。 取值范围：[0x0, 0xFFFF]
curve_para	校准曲线的系数。 取值范围：curve_para[0-5] != 0; curve_para[4] = 128。
alg_type	自动白平衡算法类型选择。 取值范围：XMEDIA_AWB_NORMAL_ALG_LOWCOST、 XMEDIA_AWB_NORMAL_ALG_ADVANCE。
rg_str	自动白平衡 R 通道校准强度。 取值范围：[0x0, 0xFF]
bg_str	自动白平衡 B 通道校准强度。

成员名称	描述
	取值范围：[0x0, 0xFF]
speed	自动白平衡算法收敛速度。 取值范围：[0x0, 0xFFFF]
zone_sel	自动白平衡算法全局或分区域计算的选择，取值范围：[0, 255]
high_ct	自动白平衡算法的色温上限。 推荐范围：[8500, 20000]
low_ct	自动白平衡算法的色温下限。 取值范围：[0, high_color_temp)
ct_limit	白平衡算法环境色温超过预设色温范围后，增益计算模式及手动增益值设定。
shift_limit_en	自动白平衡算法的 shift 偏移量开关。
shift_limit	室内场景或室内外检测关闭时，自动白平衡的白点范围参数。 取值范围：[0x0, 0xFF]
gain_norm_en	自动白平衡算法的增益归一化的开关。
natural_cast_en	低色温下 AWB 风格喜好开关。
cbcr_track	Bayer 域统计信息与 ISO 的联动参数。
luma_hist	白平衡的亮度直方图统计参数。
awb_zone_wt_en	白平衡的分块权重使能开关，默认 XMEDIA_FALSE。
zone_wt	白平衡的 1024 分块权重表，取值范围：[0x0, 0xFF]。

【注意事项】

- enable 为 XMEDIA_TRUE 时，AWB 正常工作，RGB 通道增益系数由 AWB 根据环境色温计算；enable 为 XMEDIA_FALSE 时，AWB 不工作，RGB 通道增益系数固定，为校正的静态白平衡系数。
- ref_color_temp、static_wb、curve_para 是用户通过校正工具得到的 AWB 参数，是 AWB 准确的前提。更换光学器件后，建议重新校正这三组参数。ref_color_temp 是进行静态白平衡校正光源的真实色温，推荐在 D50 光源下做静态白平衡校正。

curve_para [0], curve_para [1], curve_para [2]三个参数确定 Planckian 曲线, curve_para [3], curve_para [4], curve_para [5]三个参数确定色温曲线。

- 自动白平衡 R/B 通道校准强度, 一般情况不建议调整。

可通过调整 R/B 通道校准强度使 AWB 校准偏强或者偏弱, 0x80 表示标准强度。建议 rg_strength、bg_strength 两个参数取相同值, 且小于等于 0x80。在低色温场景, 校准强度小于 0x80 时图像偏红, 大于 0x80 时图像偏蓝; 在高色温场景, 校准强度小于 0x80 时图像偏蓝, 大于 0x80 时图像偏红。

- speed 控制 AWB 增益的收敛速度。speed 越大, 切换光源时, AWB 收敛的速度越快, 但帧间的波动幅度也变大; speed 越小, 切换光源时, AWB 收敛的速度越慢, 但帧间的波动幅度变小, 稳定性提高。

- zone_sel 参数为 0 或 255 时, 采用了近似灰度世界的算法; zone_sel 为其他值时, AWB 算法会对所有统计块进行分类筛选, 提高 AWB 精度。

- 自动白平衡算法的色温上限/下限, 以 Kelvin 为单位。

AWB 算法支持的最高/最低色温。若实际场景中的色温大于色温上限或小于色温下限, 则 AWB 不能完全恢复, 图像会偏向光源色, 低色温下偏黄, 高色温下偏蓝。

- shift_limit_en 决定 AWB 增益是否向 Planckian 曲线做映射。shift_limit 控制白色块的范围, 取值较小时, 对光源的支持范围较窄, 精度较高; 取值较大时, 支持更多的光源, 精度降低。shift_limit_en 仅控制 AWB 增益是否向 Planckian 曲线做映射, 无论 shift_limit_en 打开或关闭, shift_limit 取值的改变都会影响 AWB 结果。校正静态白平衡系数的光源色温值, 要求在 AWB 算法设置的色温上下限范围内。

- gain_norm_en 使能, 对 RGB 通道增益进行限制, 可以改善低色温、低照度场景的信噪比。

- natural_cast_en 使能, 低色温下 AWB 保留光源色, 图像颜色更自然。

- 权重表与 AWB 统计分块的对应顺序如下:

- 非拼接模式下, AWB 统计信息分块个数是 32x32, zone_wt [0]对应第一行第一列, zone_wt [1]对应第一行第二列, zone_wt [31]对应第一行第 32 列, zone_wt [32]对应第二行第一列, 依次类推。
- 不支持拼接模式下配置 AWB 统计信息权重。

- 特定应用产品, 客户使能 awb_zone_wt_en 后, 可通过设置权重表改变每个区域的权重, 优化 AWB 表现。在 Shading 较严重时, 可加大中心区域的权重, 保证中间区域 AWB 准确还原, 降低 Shading 对 AWB 的影响。在行车记录仪应用, 感兴趣区域一般在画面的中心偏下区域, 可加大此区域的权重, 降低天空、树木等区域对 AWB 的干扰。

【相关数据类型及接口】

xmedia_awb_attr

xmedia_awb_attr

【说明】

定义白平衡属性。

【定义】

```
typedef struct {  
    xmedia_bool          bypass;  
    xmedia_u8            awb_runinterval;  
    xmedia_video_operation_mode op_type;  
    xmedia_awb_alg        alg_type;  
    xmedia_awb_manual_attr manual_attr;  
    xmedia_awb_auto_attr  auto_attr;  
} xmedia_awb_attr;
```

【成员】

成员名称	描述
bypass	白平衡模块 Bypass 使能，默认值 XMEDIA_FALSE。
awb_runinterval	白平衡模块工作频率。取值范围：[0x1, 0xFF]；
op_type	自动白平衡和手动白平衡切换。
alg_type	算法类型选择。
manual_attr	手动白平衡参数。
auto_attr	自动白平衡参数。

【注意事项】

- bypass 为 XMEDIA_TRUE 时，WB 的其它参数设置不生效，RGB 通道增益系数固定为 0x100。
- awb_runinterval 默认为 4，表示每 4 帧运行一次 AWB。客户可根据业务需求降低 AWB 的运行频率。
- op_type 默认为 alg_type 代表使用原有的 AWB 算法，用户可以使用 ALG_AWB_AI 切换为机器学习 AWB 算法。

【相关数据类型及接口】

[xmedia_awb_set_attr](#)

[xmedia_awb_get_attr](#)

xmedia_awb_extra_lightsource_info

【说明】

定义独立光源点的信息。

【定义】

```
typedef struct {  
    xmedia_u16 white_rgain;  
    xmedia_u16 white_bgain;  
    xmedia_u16 exp_quant;  
    xmedia_u16 light_status;  
    xmedia_u16 radius;  
} xmedia_awb_extra_lightsource_info;
```

【成员】

成员名称	描述
white_rgain	增加或删除区域的中心 Cr 坐标，8bit 小数精度。 取值范围：[0x0, 0xFFF]
white_bgain	增加或删除区域的中心 Cb 坐标，8bit 小数精度。 取值范围：[0x0, 0xFFF]
exp_quant	对外界环境亮度做判断，暂未支持，不需要配置。
light_status	光源点状态。 0：不使能； 1：增加光源点； 2：删除干扰色；
radius	增加或删除区域的半径。 取值范围：[0x0, 0xFF]

【注意事项】

- 独立光源点实现以下功能：用户设定 CbCr 坐标的某个区域，指定该区域为特殊光源下的白色表现，增加为独立光源点；或指定该区域为特定光源下的干扰色，进行删除。
- 通过 light_status 参数的设置实现功能选择。light_status 设置为 1 时，在用户校正的 Plankian 曲线外，增加特殊光源点，以优化该光源下颜色表现，不影响设备在其它光源下的表现；light_status 设置为 2 时，支持用户删除特定光源的某个干扰色(肤色、绿色、蓝天等)。
- 建议删除干扰色仅用来优化特定环境的颜色表现，否则会影响其它光源的表现。在 CbCr 坐标，5500K 光源下的肤色与 3500K 光源的白色重合度很高，在 5500K 环境删除肤色的干扰，会导致 3500K 光源下偏色。
- white_rgain、white_bgain 的取值。在该光源下捕获 ColorChecker Raw 数据，通过 Calibration Tool Static WB 选项校准得到的 Rgain、Bgain 即 white_rgain、white_bgain 坐标值。删除干扰色，可选择干扰色区域进行静态白平衡校正，来确定 white_rgain、white_bgain 坐标值。
- radius 取值由用户设置，增加光源点，建议取值在[0x8, 0x10]范围内，删除干扰色，尽量避免[white_rgain、white_bgain]为中心，radius 为半径的圆形区域与 Plankian 曲线重合。增加光源点或删除干扰色，设置 radius 半径太大时(覆盖场景内大多数统计块)，会导致功能失效，甚至起反作用。
- 当前支持定义最多 4 个光源色，4 个相互独立。4 个光源点需要避免交集，特别需要避免以下情况：CbCr 坐标的单个点，既指定为增加的独立光源点，又指定为删除的干扰色。

【相关数据类型及接口】

无

xmedia_awb_indoor_outdoor_status

【说明】

定义 AWB 室内外状态。

【定义】

```
typedef enum {
    XMEDIA_AWB_INDOOR_MODE = 0,
    XMEDIA_AWB_OUTDOOR_MODE = 1,
    XMEDIA_AWB_INDOOR_OUTDOOR_MAX
} xmedia_awb_indoor_outdoor_status;
```

【成员】

成员名称	描述
XMEDIA_AWB_INDOOR_MODE	室内模式。
XMEDIA_AWB_OUTDOOR_MODE	室外模式。

【注意事项】

无

【相关数据类型及接口】

[xmedia_awb_in_out_attr](#)

[xmedia_awb_scene_adj_attr](#)

xmedia_awb_in_out_attr

【说明】

定义对场景做室内外判断的参数。

【定义】

```
typedef struct {
    xmedia_bool          enable;
    xmedia_video_operation_mode op_type;
    xmedia_awb_indoor_outdoor_status outdoorstatus;
    xmedia_u32           out_thresh;
    xmedia_u16           low_start;
    xmedia_u16           low_stop;
    xmedia_u16           high_start;
    xmedia_u16           high_stop;
    xmedia_bool          green_enhanceen;
    xmedia_u8            out_shift_limit;
} xmedia_awb_in_out_attr;
```

【成员】

成员名称	描述
enable	对场景做室内外判断开关。
op_type	对场景做室内外判断类型(自动或手动)。
outdoorstatus	手动模式下，用户输入室内外状态信息。

成员名称	描述
out_thresh	室内外场景判定设置的阈值，以单位为 us 的曝光时间，小于该阈值则认为是室外。
low_start	将位于低色温范围内的像素权重调低，低色温区起始值，建议 5000K。
low_stop	将位于低色温范围内的像素权重调低，低色温区终止值，建议 4500K。 取值范围：(0, low_start)
high_start	将位于高色温范围内的像素权重调低，高色温区起始值，建议 6500K。 取值范围：(low_start, 0xFFFF]
high_stop	将位于高色温范围内的像素权重调低，高色温区终止值，建议 8000K。 取值范围：(high_start, 0xFFFF]
green_enhanceen	在绿色植物场景下是否对绿色通道增强。
out_shift_limit	判断为室外场景时，AWB 算法的白点范围限制 Shift=out_shift_limit。 取值范围：[0x0, 0xFF]

【注意事项】

- out_thresh 需要根据 sensor 的感光特性进行调整。等效曝光时间=曝光时间*模拟增益*数字增益*ISP 数字增益。
- 用户调用自己的 AE 库，且室内外判断类型为自动时，请务必将当前曝光时间和 ISO 传递给 AWB 库。
- 设置色温范围的 low_start、low_stop、high_start、high_stop 等 4 个参数，和 sensor 校正相关，建议用户针对 sensor 做微调。四个参数的取值范围要求：low_stop<low_start<high_start<high_stop。如果设置的室外色温范围[low_start, high_start]较宽，室外的大面积纯色场景颜色表现可能会稍差；如果设置的室外色温范围[low_start, high_start]较窄，在日落前等高低色温场景，AWB 不能完全恢复，低色温下稍微偏黄，高色温下稍微偏蓝。

- AWB 校正参数调整后，low_start、low_stop、high_start、high_stop 等 4 个参数要再次进行调整，以达到最佳效果。
- Shift 取值影响 AWB 算法表现。普通场景 shift_limit、out_shift_limit 两个参数的取值大小对 AWB 结果影响不大。特殊光源、大面积单色场景，shift_limit、out_shift_limit 两个参数的取值大小对 AWB 结果影响较大。两个 Shift 参数取值较小时，单色表现好，但特殊光源下容易偏色；两个 Shift 参数取值较大时，AWB 表现稳定性好，但大面积绿色、黄色等场景可能偏色。

【相关数据类型及接口】

[xmedia_awb_set_scene_adj_attr](#)

[xmedia_awb_get_scene_adj_attr](#)

xmedia_awb_multi_ls_type

【说明】

定义混合光源下的 AWB 策略。

【定义】

```
typedef enum {
    XMEDIA_AWB_MULTI_LS_SAT = 0,
    XMEDIA_AWB_MULTI_LS_CCM = 1,
    XMEDIA_AWB_MULTI_LS_MAX
} xmedia_awb_multi_ls_type;
```

【成员】

成员名称	描述
XMEDIA_AWB_MULTI_LS_SAT	混合光源下自动调整饱和度。
XMEDIA_AWB_MULTI_LS_CCM	混合光源下自动调整 CCM。

【注意事项】

- XMEDIA_AWB_MULTI_LS_CCM 对图像饱和度损失较少，但会改变颜色的色调。
- XMEDIA_AWB_MULTI_LS_SAT 方式降低图像整体饱和度，改善混合光源下的偏色程度。

【相关数据类型及接口】[xmedia_awb_scene_adj_attr](#)

[xmedia_awb_set_scene_adj_attr](#)

[xmedia_awb_get_scene_adj_attr](#)

xmedia_awb_scene_adj_attr

【说明】

awb 场景调优属性，用于某些特殊场景(室外、混合光源、肤色/蓝天增强场景) 参数调优。

【定义】

```
typedef struct {  
    xmedia_u8  tolerance;  
    xmedia_u8  iso_num;  
    xmedia_u32 iso_level[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u8  zone_radius[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u16 curve_l_limit;  
    xmedia_u16 curve_r_limit;  
    xmedia_bool extra_lighten;  
    xmedia\_awb\_extra\_lightsource\_info light_info[XMEDIA_ISP_AWB_LS_NUM];  
    xmedia\_awb\_in\_out\_attr in_or_out;  
    xmedia_bool multilight_source_en;  
    xmedia\_awb\_multi\_ls\_type multi_ls_type;  
    xmedia_u16 multi_ls_scaler;  
    xmedia_u16 multi_ct_bin[XMEDIA_ISP_AWB_MULTI_CT_NUM];  
    xmedia_u16 multi_ct_wt[XMEDIA_ISP_AWB_MULTI_CT_NUM];  
    xmedia_bool fine_tun_en;  
    xmedia_u8  fine_tun_strength;  
} xmedia_awb_scene_adj_attr;
```



说明

宏定义如下：

- #define XMEDIA_ISP_AWB_LS_NUM 4
- #define XMEDIA_ISP_AWB_MULTI_CT_NUM 6

【成员】

成员名称	描述
tolerance	自动白平衡调整的偏差范围，检测误差在门限范围内时，AWB 不动作，取值范围[0,255]
iso_num	zone_radius iso 个数。
iso_level	zone_radius iso 等级。
zone_radius	自动白平衡统计中对像素分类时用的距离范围。

成员名称	描述
	zone_radius 越小，靠近普朗克曲线的白点权重越大，取值范围[0,255]
curve_l_limit	自动白平衡色温曲线的左下侧界限，取值范围[0,256]
curve_r_limit	自动白平衡色温曲线的右上侧界限，取值范围[256,4095]
extra_lighten	其他光源点使能。
light_info	其他光源点位置信息。
in_or_out	自动白平衡对场景做室内外判断的参数。
multilight_source_en	混合光源模式使能。
multi_ls_type	混合光源调整策略选择，支持调整饱和度或 CCM。
multi_ls_scaler	混合光源下，饱和度或 CCM 最大调整幅度。实际调整幅度还和场景混合光源程度有关。取值范围[0,256]。
multi_ct_bin	混合光源下的色温分段参数，取值范围[0,65535]。
multi_ct_wt	混合光源下的色温权重参数，取值范围[0,1024]。
fine_tun_en	特殊色增强开关。
fine_tun_strength	128 位基础倍数， ≥ 128 为肤色增强， ≤ 128 为蓝天增强 算法会自动检测特殊色比重，自适应调节增强权重，范围为[0, 255]。

【注意事项】

无

【相关数据类型及接口】

[xmedia_awb_set_scene_adj_attr](#)

[xmedia_awb_get_scene_adj_attr](#)

xmedia_awb_ccm_manual

【说明】

定义手动颜色校正矩阵属性。

【定义】

```
typedef struct {  
    xmedia_bool sat_en;  
    xmedia_u16 ccm[XMEDIA_ISP_CCM_MATRIX_SIZE];  
} xmedia_awb_ccm_manual;
```

【成员】

成员名称	描述
sat_en	手动 CCM 模式下，饱和度是否生效。
ccm	手动颜色校正矩阵，8bit 小数精度。bit 15 是符号位，0 表示正数，1 表示负数，例如 0x8010 表示-16。 取值范围：[0x0, 0xFFFF]

【注意事项】

手动 CCM 模式下，用户可通过 sat_en 参数选择饱和度是否生效。

- sat_en = 1，生效的 CCM = Manual_CCM * Saturation_CCM，用户可用该功能调节手动 ccm 下，不同照度下的饱和度强度；
- sat_en = 0，生效的 CCM = Manual_CCM。

sat_en = 1，生效的 CCM 会自动修正，保持每行参数的和为 0x100。sat_en = 0，生效的 CCM 为手动设定参数值，不进行 CCM 每行求和 0x100 的修正。

【相关数据类型及接口】

[xmedia_awb_ccm_attr](#)

[xmedia_awb_set_ccm_attr](#)

[xmedia_awb_ccm_param](#)

【说明】

定义不同色温下自动颜色校正矩阵系数。

【定义】

```
typedef struct {  
    xmedia_u16 ct;  
    xmedia_u16 ccm[XMEDIA_ISP_CCM_MATRIX_SIZE];  
} xmedia_awb_ccm_param;
```

【成员】

成员名称	描述
ct	当前配置的 CCM 对应的色温。 取值范围：[500, 30000]
ccm	不同色温下的颜色校正矩阵，8bit 小数精度。bit 15 是符号位，0 表示正数，1 表示负数，例如 0x8010 表示-16。bit12,13,14 是无效的，即正数从 0x0000 到 0x0FFF，负数从 0x8000 到 0x8FFF，是 CCM 的合理参数。 取值范围：[0x0,0xFFFF]

【注意事项】

- 颜色校正矩阵的数据格式，应与校正工具提供的保持一致。
- 当饱和度为自动的时候，实际生效的 CCM 参数和 AWB 算法反馈的当前色温值，自动饱和度参数有关。当饱和度为手动的时候，实际生效的 CCM 参数和 AWB 算法反馈的当前色温值有关。

【相关数据类型及接口】

[xmedia_awb_ccm_auto](#)

xmedia_awb_ccm_auto

【说明】

定义自动颜色校正矩阵属性。

【定义】

```
typedef struct {  
    xmedia_bool      iso_act_en;  
    xmedia_bool      temp_act_en;  
    xmedia_u16        ccm_tab_num;  
    xmedia\_awb\_ccm\_param ccm_tab[XMEDIA_ISP_CCM_MATRIX_NUM];  
} xmedia_awb_ccm_auto;
```



说明

```
#define XMEDIA_ISP_CCM_MATRIX_NUM
```

7

【成员】

成员名称	描述
iso_act_en	是否使能低照度下 CCM Bypass 功能。
temp_act_en	是否使能高、低色温下 CCM Bypass 功能。
ccm_tab_num	当前配置的 CCM 的组数。 取值范围：[3,7]
ccm_tab	不同色温下的颜色校正矩阵和对应的色温值。

【注意事项】

- 当 xmedia_awb_saturation_attr 中的 op_type 为 XMEDIA_VIDEO_OPERATION_MODE_AUTO，使用自动饱和度调节功能时，iso_act 和 temp_act 功能生效。当 op_type 为 XMEDIA_VIDEO_OPERATION_MODE_MANUAL，使用手动饱和度调节功能时，iso_act_en 和 temp_act_en 功能不生效。
- iso_act 功能指的是极低照度下（ISO 大于 12800）逐渐 Bypass CCM，将由饱和度数组 sat [16]控制得到的 CCM 矩阵逐渐调向接近单位阵，这时 CCM 的实际饱和度和饱和度数组中设定的不一致。用户关闭该功能后，极低照度下，饱和度数组仍生效。
- temp_act 功能指的是在高色温(大于 8000K)和低色温场景(小于 2500K)，AWB 可能未完全恢复，此时采用高饱和度 CCM 矩阵，会加剧 AWB 的偏色程度。比如钠灯场景，AWB 校正后白色区域蓝色稍有不足，采用高饱和度 CCM 矩阵，白色区域的蓝色分量可能为 0，图像整体偏黄。使能 temp_act_en，在高低色温下逐渐 BypassCCM，将由饱和度数组 sat [16]控制得到的 CCM 矩阵逐渐调向接近单位阵，优化类似场景颜色表现。
- 颜色校正矩阵的数据格式，应与校正工具提供的保持一致。

【相关数据类型及接口】

[xmedia_awb_ccm_attr](#)

xmedia_awb_ccm_attr

【说明】

定义 ISP 颜色矩阵属性。

【定义】

```
typedef struct {
    xmedia_bool      ccm_en;
    xmedia_awb_ccm_manual manual_attr;
    xmedia_awb_ccm_auto  auto_attr;
} xmedia_awb_ccm_attr;
```

【成员】

成员名称	描述
ccm_en	ccm 使能开关。
manual_attr	手动颜色校正矩阵。
auto_attr	自动颜色校正矩阵。

【注意事项】

无。

【相关数据类型及接口】

[xmedia_awb_set_ccm_attr](#)

[xmedia_awb_get_ccm_attr](#)

xmedia_awb_saturation_manual

【说明】

定义手动饱和度属性。

【定义】

```
typedef struct {
    xmedia_u8 saturation;
} xmedia_awb_saturation_manual;
```

【成员】

成员名称	描述
saturation	手动饱和度值。 取值范围：[0x0, 0xFF]

【注意事项】

无。

【相关数据类型及接口】

[xmedia_awb_saturation_attr](#)

xmedia_awb_saturation_auto

【说明】

定义自动饱和度属性。

【定义】

```
typedef struct {  
    xmedia_u8 iso_num;  
    xmedia_u32 iso_level[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u8 sat[XMEDIA_ISP_ISO_MAX_COUNT];  
} xmedia_awb_saturation_auto;
```

【成员】

成员名称	描述
iso_num	ISO 数量。
iso_level	ISO 等级。
sat	自动饱和度值。 推荐配置为递减序列。

【注意事项】

无。

【相关数据类型及接口】

[xmedia_awb_saturation_attr](#)

xmedia_awb_saturation_attr

【说明】

定义 ISP 颜色饱和度属性。

【定义】

```
typedef struct {
```

```

xmedia_video_operation_mode      op_type;
xmedia_awb_saturation_manual manual_attr;
xmedia_awb_saturation_auto      auto_attr;
} xmedia_awb_saturation_attr;

```

【成员】

成员名称	描述
op_type	设置饱和度类型(手动或自动)。
manual_attr	手动饱和度参数。
auto_attr	自动饱和度参数。

【注意事项】

- 设置饱和度功能分为自动和手动：
 - op_type 为 XMEDIA_VIDEO_OPERATION_MODE_AUTO，使用自动饱和度调节功能。
此时饱和度值会根据系统增益自动调节，饱和度与系统增益的关系请参见成员变量 sat [16]的描述。
在自动饱和度模式下，系统最终的饱和度由系统增益对应的 sat [16]变量，AWB 控制的混合光源下自动调节饱和度 XMEDIA_AWB_MULTI_LS_SAT，极低照度下 CCM Bypass 功能 ISOAct，高、低色温下 CCM Bypass 功能 temp_act，几个因素共同决定。
 - op_type 为 XMEDIA_VIDEO_OPERATION_MODE_MANUAL，使用手动饱和度调节功能。
此时变量 sat[16]无效。
在手动饱和度模式下，系统最终的饱和度由用户完全控制。

【相关数据类型及接口】

[xmedia_awb_set_saturation_attr](#)
[xmedia_awb_get_saturation_attr](#)

xmedia_awb_info

【说明】

定义白平衡，饱和度，颜色校正信息。

【定义】

```
typedef struct {  
    xmedia_u16 rgain;  
    xmedia_u16 grgain;  
    xmedia_u16 gbgain;  
    xmedia_u16 bgain;  
    xmedia_u16 saturation;  
    xmedia_u16 ct;  
    xmedia_u16 ccm[XMEDIA_ISP_CCM_MATRIX_SIZE];  
    xmedia_u16 ls0_ct;  
    xmedia_u16 ls1_ct;  
    xmedia_u16 ls0_area;  
    xmedia_u16 ls1_area;  
    xmedia_u8 multi_degree;  
    xmedia_u16 active_shift;  
    xmedia_u32 first_stable_time;  
    xmedia_awb_indoor_outdoor_status inout_status;  
    xmedia_s16 bv;  
} xmedia_awb_info;
```

【成员】

成员名称	描述
rgain	当前 R 通道增益值，8bit 小数精度。
grgain	当前 Gr 通道增益值，8bit 小数精度。
gbgain	当前 Gb 通道增益值，8bit 小数精度。
bgain	当前 B 通道增益值，8bit 小数精度。
saturation	当前饱和度值，有效范围为[0, 255]。
ct	当前色温值。
ccm	当前颜色校正矩阵值，8bit 小数精度。bit 15 是符号位，0 表示正数，1 表示负数，例如 0x8010 表示-16。
ls0_ct	混合光源场景，主光源色温。
ls1_ct	混合光源场景，次要光源色温。
ls0_area	混合光源场景，主光源面积。取值范围为[0x0, 0xFF]。

成员名称	描述
ls1_area	混合光源场景，次要光源面积。取值范围为[0x0, 0xFF]。
multi_degree	当前场景是混合光源的概率。
active_shift	当前场景最终生效的白点范围限制 shift 值。取值范围为 [0x0,0xFF]。
first_stable_time	首次 AWB 收敛稳定的时间，以帧为计数单位。
inout_status	室内外检测结果。
bv	当前环境曝光量。

【注意事项】

- multi_degree 非 0 时，表示场景是混合光源。该值越大，场景为混合光源的概率越大。
- ls0_ct、ls1_ct、ls0_area、ls1_area 等查询结果仅在 multi_degree 非 0 时有效。
- 混合光源检测仅在室内场景打开，AWB 判断为室外场景后，逐渐减弱混合光源检测的强度，以避免饱和度降低。
- 混合光源概率说明：主光源和次要光源的亮度、色温差等条件影响混合光源的概率计算。检测到场景内存在主光源和次要光源两种光源后，主光源和次要光源的色温差异越大、亮度差异越小，混合光源概率越大。
- ls0_area 和 ls1_area 之和有可能大于白平衡分区个数，原因是部分区域可能受到两种光源的影响，算法不将重叠区域严格识别，会叠加到主光源、次要光源。
- first_stable_time 是 AWB 首次收敛稳定后所对应的帧数，非 0 表示 AWB 已稳定。ISP 系统启动后，1-128 帧 AWB 固定每帧执行一次，以加速 AWB 收敛。

【相关数据类型及接口】

[xmedia_awb_query_info](#)

xmedia_awb_color_tone_attr

【说明】

ColorTone 提供色调调节接口，用户可通过该接口实现图像颜色偏红、偏绿、偏蓝等喜好调节。

【定义】

```
typedef struct {
    xmedia_u16 red_cast_gain;
    xmedia_u16 green_cast_gain;
    xmedia_u16 blue_cast_gain;
} xmedia_awb_color_tone_attr;
```

【成员】

成员名称	描述
red_cast_gain	手动 RGain 增强强度，取值范围[256,384]。
green_cast_gain	手动 GGain 增强强度，取值范围[256,384]。
blue_cast_gain	手动 BGain 增强强度，取值范围[256,384]。

【注意事项】

无

【相关数据类型及接口】

[xmedia_awb_set_color_tone_attr](#)

[xmedia_awb_get_color_tone_attr](#)

xmedia_color_sector_manual

【说明】

定义手动颜色分区调节功能属性。

【定义】

```
typedef struct {
    xmedia_u16 hue_shift[XMEDIA_ISP_COLOR_SECTOR_NUM];
    xmedia_u16 sat_shift[XMEDIA_ISP_COLOR_SECTOR_NUM];
} xmedia_color_sector_manual;
```



说明

宏定义如下：

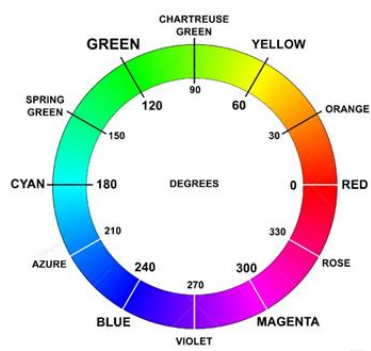
```
#define XMEDIA_ISP_COLOR_SECTOR_NUM 6
```

【成员】

成员名称	描述
hue_shift	特定颜色色调角度调整，6 个数组分别对应颜色：红、肤、

成员名称	描述
	黄、绿、蓝、紫，如下图所示， 20 为不调整，大于 20 为顺时针偏移，小于 20 为逆时针偏移
sat_shift	饱和度调整，6 个数组分别对应颜色：红、肤、黄、绿、蓝、紫，如下图所示， 20 为不调整，大于 20 为饱和度增大，小于 20 为饱和度减小

图6-2 红、肤、黄、绿、蓝、紫对应的角度



【注意事项】

XM7606V13 暂未支持。

【相关数据类型及接口】

[xmedia_color_sector_attr](#)

xmedia_color_sector_auto

【说明】

定义自动颜色分区调节功能属性。

【定义】

```
typedef struct {
    xmedia_u8  ct_num;
    xmedia_u32 ct_level[XMEDIA_ISP_CCM_TAB_MAX_NUM];
    xmedia_u16
    hue_shift[XMEDIA_ISP_CCM_TAB_MAX_NUM][XMEDIA_ISP_COLOR_SECTOR_NUM];
    xmedia_u16
    sat_shift[XMEDIA_ISP_CCM_TAB_MAX_NUM][XMEDIA_ISP_COLOR_SECTOR_NUM];
} xmedia_color_sector_auto;
```

【成员】

成员名称	描述
ct_num	色温数量。取值范围[1,7]
ct_level	色温等级。
hue_shift	7 个色温区间色调角度调整，6 个数组分别对应颜色：红、肤、黄、绿、蓝、紫，取值范围[0,40] 20 为不调整，大于 20 为顺时针偏移，小于 20 为逆时针偏移
sat_shift	7 个色温区间饱和度调整，6 个数组分别对应颜色：红、肤、黄、绿、蓝、紫，取值范围[0,40] 20 为不调整，大于 20 为饱和度增大，小于 20 为饱和度减小

【注意事项】

XM7606V13 暂未支持。

【相关数据类型及接口】

[xmedia_color_sector_attr](#)

xmedia_color_sector_attr

【说明】

定义颜色分区调节功能属性。

【定义】

```
typedef struct {  
    xmedia_bool          sector_en;  
    xmedia\_video\_operation\_mode op_type;  
    xmedia\_color\_sector\_manual  manual_attr;  
    xmedia\_color\_sector\_auto    auto_attr;  
} xmedia_color_sector_attr;
```

【成员】

成员名称	描述
sector_en	颜色分区调节开关。
op_type	设置颜色分区调节类型(手动或自动)。
manual_attr	手动颜色分区调节参数。
auto_attr	自动颜色分区调节参数。

【注意事项】

- 设置颜色分区调节功能分为自动和手动：

- op_type 为 XMEDIA_VIDEO_OPERATION_MODE_AUTO，使用自动颜色分区调节功能。

此时 hue_shift、sat_shift 会根据 AWB 计算色温值自动调节强度。

- op_type 为 XMEDIA_VIDEO_OPERATION_MODE_MANUAL，使用手动颜色分区调节功能。

在手动颜色分区调节模式下，系统最终的 hue_shift、sat_shift 由用户完全控制。

XM7606V13 暂未支持。

【相关数据类型及接口】

[xmedia_awb_set_color_sector_attr](#)

[xmedia_awb_get_color_sector_attr](#)

7 AF

7.1 概述

AF 利用图像的对比度或相位信息，计算 lens 驱动的位移量，使得焦平面尽量与 sensor 平面重合，达到 sensor 清晰成像的目的。

7.2 重要概念

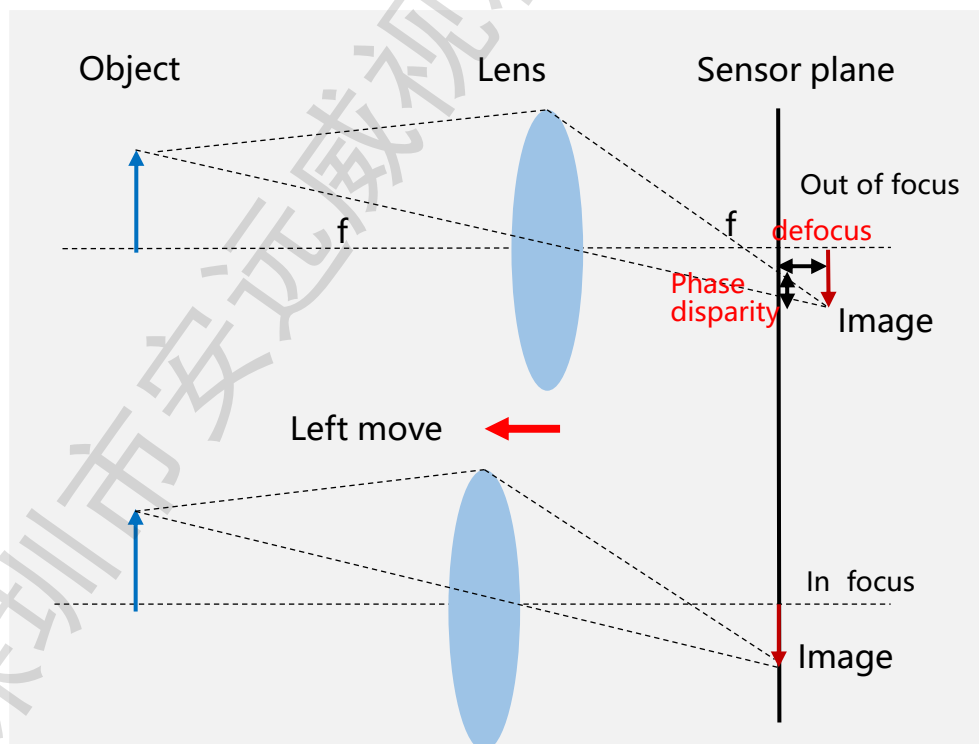
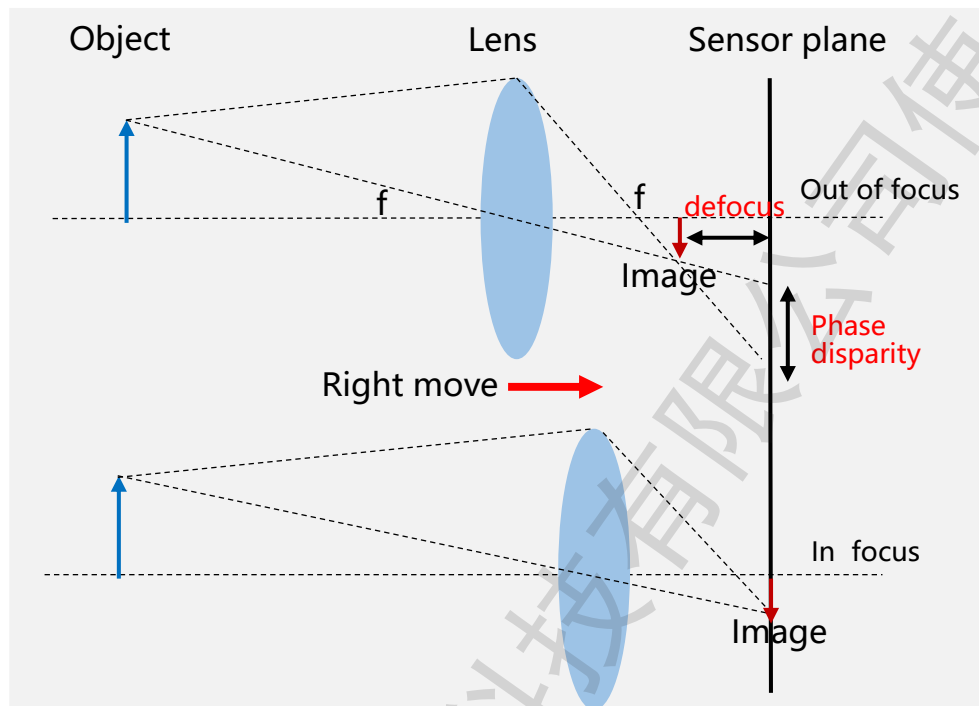
PDAF：相位检测自动对焦。

CDAF：反差检查自动对焦。

7.3 功能描述

PDAF 首先利用 sensor 中相位检测像素计算得到相位差（Phase Disparity, PD）信息，然后将 PD 信息转换为镜头需要移动的失焦量（Defocus），最终实现画面中感兴趣区域（Region of Interest, ROI）的快速对焦，对焦示意图如图 7-1 所示。

图7-1 PDAF 对焦示意图 (分别为焦后与焦前)



7.4 API 参考

7.4.1 AF 库接口

所有 AF 库接口都只是针对 XMEDIA_AF 库，如果客户自己实现 AF 库，不需要关注这些接口，且无法使用这些接口。

- `xmedia_af_register`: 向 ISP 注册 AF 库。
- `xmedia_af_unregister`: 向 ISP 反注册 AF 库。

`xmedia_af_register`

【描述】

向 ISP 注册 AF 库。

【语法】

```
xmedia_s32 xmedia_af_register(xmedia_u32 pipe);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: `xmedia_af.h`
- 库文件: `libxmedia_isp.a`、`libxmedia_af.a`

【注意】

- 该接口调用了 ISP 库提供的 AF 注册回调接口 xmedia_isp_register_af_lib，以实现向 ISP 库注册的功能。
- 用户调用此接口完成 XMEDIA_AF 库向 ISP 库注册。
- AF 库可以注册多个实例，此接口不支持多进程操作。

【举例】

无

【相关主题】

[xmedia_af_unregister](#)

xmedia_af_unregister

【描述】

向 ISP 反注册 AF 库。

【语法】

```
xmedia_s32 xmedia_af_unregister(xmedia_u32 pipe);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_af.h
- 库文件：libxmedia_isp.a、libxmedia_af.a

【注意】

- 该接口调用了 ISP 库提供的 AF 反注册回调接口。xmedia_isp_unregister_af_lib, 以实现 AF 向 ISP 库反注册的功能。
- 用户调用此接口完成 XMEDIA_AF 库向 ISP 库反注册, 此接口不支持多进程操作。

【举例】

无

【相关主题】

[xmedia_af_register](#)

7.4.2 AF 控制模块

- [xmedia_af_set_attr](#): 设置 AF 属性。
- [xmedia_af_get_attr](#): 获取 AF 属性。
- [xmedia_af_query_info](#): 查询 AF 状态信息。

xmedia_af_set_attr

【描述】

设置 AF 属性。

【语法】

```
xmedia_s32 xmedia_af_set_attr(xmedia_u32 pipe, xmedia\_af\_attr *af_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
af_attr	AF 的参数属性结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败, 其值为错误码。

【需求】

- 头文件: xmedia_af.h
- 库文件: libxmedia_isp.a、libxmedia_af.a

【举例】

无

【相关主题】

[xmedia_af_get_attr](#)

xmedia_af_get_attr

【描述】

获取 AF 属性。

【语法】

```
xmedia_s32 xmedia_af_get_attr(xmedia_u32 pipe, xmedia\_af\_attr *af_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
af_attr	AF 的参数属性结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_af.h
- 库文件: libxmedia_isp.a、libxmedia_af.h

【注意】

无

【相关主题】

[xmedia_af_set_attr](#)

xmedia_af_query_info

【描述】

查询 AF 状态信息。

【语法】

```
xmedia_s32 xmedia_af_query_info(xmedia_u32 pipe, xmedia\_af\_info *af_info);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
af_info	AF 的状态信息结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_af.h
- 库文件: libxmedia_isp.a、libxmedia_af.h

【注意】

无

【相关主题】

[xmedia_af_set_attr](#)

7.5 数据类型

7.5.1 Register

- [xmedia_lens_register_af_info](#): 定义 lens 驱动与 AF 库之间接口结构体。
- [xmedia_lens_register_af_func](#): 定义 lens 回调函数结构体。

xmedia_lens_register_af_info

【说明】

定义 lens 驱动与 AF 库之间接口结构体。

【定义】

```
typedef struct {
    xmedia_u32          dev_id;
    xmedia\_lens\_register\_af\_func func;
} xmedia_lens_register_af_info;
```

【成员】

成员名称	描述
dev_id	lens 设备号。
func	lens 库提供的回调函数结构体。

【差异说明】

无。

【注意事项】

无。

【相关数据类型及接口】

[xmedia_lens_register_af_func](#)

xmedia_lens_register_af_func

【说明】

定义 lens 回调函数结构体。

【定义】

```
typedef struct {
    xmedia_s32 (*pfn_lens_zoom_in)(xmedia_u32 dev, xmedia_u32 distance);
    xmedia_s32 (*pfn_lens_zoom_out)(xmedia_u32 dev, xmedia_u32 distance);
    xmedia_s32 (*pfn_lens_zoom_pi)(xmedia_u32 dev, xmedia_u32 pi);
    xmedia_s32 (*pfn_lens_zoom_status)(xmedia_u32 dev);
    xmedia_s32 (*pfn_lens_focus_near)(xmedia_u32 dev, xmedia_u32 distance);
    xmedia_s32 (*pfn_lens_focus_far)(xmedia_u32 dev, xmedia_u32 distance);
    xmedia_s32 (*pfn_lens_focus_pi)(xmedia_u32 dev, xmedia_u32 pi);
    xmedia_s32 (*pfn_lens_focus_status)(xmedia_u32 dev);
} xmedia_lens_register_af_func;
```

【成员】

成员名称	描述
pfn_lens_zoom_in	变倍放大的回调函数指针。
pfn_lens_zoom_out	变倍缩小的回调函数指针。
pfn_lens_zoom_pi	变倍镜头回到 pi 点的回调函数指针。
pfn_lens_zoom_status	获取变倍镜头的工作状态的回调函数指针。
pfn_lens_focus_near	对焦镜头向 near 端移动的回调函数指针。
pfn_lens_focus_far	对焦镜头向 far 端移动的回调函数指针。
pfn_lens_focus_pi	对焦镜头回到 pi 点的回调函数指针。
pfn_lens_focus_status	获取对焦镜头的工作状态的回调函数指针。

【注意事项】

如果回调函数指针不需要赋值，需要置为 NULL。

【相关数据类型及接口】

[xmedia_lens_register_af_info](#)

7.5.2 AF

- [xmedia_af_attr](#): 定义 AF 属性。
- [xmedia_af_type](#): 定义 AF 类型。
- [xmedia_af_pdaf_attr](#): 定义 PDAF 属性。
- [xmedia_af_cdaf_attr](#): 定义 CDAF 属性。
- [xmedia_af_info](#): 定义 AF 状态信息。

xmedia_af_attr

【说明】

定义 af 属性。

【定义】

```
typedef struct {
    xmedia_bool      enable;
    xmedia_af_type    type;
    xmedia_video_rect roi;
    xmedia_u8         run_interval;
    xmedia_af_pdaf_attr pdaf_attr;
    xmedia_af_cdaf_attr cdaf_attr;
} xmedia_af_attr;
```

【成员】

成员名称	描述
enable	af 使能开关。
type	对焦类型。
roi	感兴趣区域。
run_interval	运行间隔。
pdaf_attr	相位对焦的属性。
cdaf_attr	对比度对焦的属性。

【注意事项】

无

【相关数据类型及接口】

[xmedia_af_set_attr](#)

[xmedia_af_get_attr](#)

xmedia_af_type

【说明】

定义 af 类型。

【定义】

```
typedef enum {  
    XMEDIA_AF_TYPE_CDAF,  
    XMEDIA_AF_TYPE_PDAF,  
    XMEDIA_AF_TYPE_HYBRID,  
    XMEDIA_AF_TYPE_MAX,  
} xmedia_af_type;
```

【成员】

成员名称	描述
XMEDIA_AF_TYPE_CDAF	CDAF 对焦模式。
XMEDIA_AF_TYPE_PDAF	PDAF 对焦模式。
XMEDIA_AF_TYPE_HYBRID	混合对焦模式。

【注意事项】

无

【相关数据类型及接口】

[xmedia_af_set_attr](#)

[xmedia_af_get_attr](#)

[xmedia_af_pdaf_attr](#)

【说明】

定义 pdaf 类型。

【定义】

```
typedef struct {  
    xmedia_bool vertical_binning_en;  
    xmedia_u32   vertical_binning_times;  
    xmedia_bool xcorr_smooth_en;  
    xmedia_s32  xcorr_calc_upper_thd;  
    xmedia_s32  xcorr_calc_lower_thd;  
    xmedia_u32  luma_thd;  
    xmedia_s32  phase_disparity_thd;  
    xmedia_s32  dcc_slope_adj_coeff;  
    xmedia_s32  confidence_thd_low;
```

```

xmedia_s32 confidence_thd_high;

xmedia_bool spc_en;
xmedia_u32
left_spc_gain[XMEDIA_AF_SPC_GAIN_MAP_Y_DIR_NUM][XMEDIA_AF_SPC_GAIN_MAP_X_DIR_NUM];
xmedia_u32
right_spc_gain[XMEDIA_AF_SPC_GAIN_MAP_Y_DIR_NUM][XMEDIA_AF_SPC_GAIN_MAP_X_DIR_NUM];
xmedia_s32
dcc_slope[XMEDIA_AF_DCC_MAP_Y_DIR_NUM][XMEDIA_AF_DCC_MAP_X_DIR_NUM];
xmedia_s32
dcc_offset[XMEDIA_AF_DCC_MAP_Y_DIR_NUM][XMEDIA_AF_DCC_MAP_X_DIR_NUM];
} xmedia_af_pdaf_attr;

```



说明

宏定义如下：

```

#define XMEDIA_AF_SPC_GAIN_MAP_X_DIR_NUM 8
#define XMEDIA_AF_SPC_GAIN_MAP_Y_DIR_NUM 6
#define XMEDIA_AF_DCC_MAP_X_DIR_NUM      8
#define XMEDIA_AF_DCC_MAP_Y_DIR_NUM      6

```

【成员】

成员名称	描述
vertical_binning_en	垂直 binning 使能开关。 取值范围：[0,1]。
vertical_binning_times	垂直 binning 行数，值越大，能提高降噪效果和计算效率，但可能会丢失相位信息。 取值范围：[0,32]。
xcorr_smooth_en	相关计算平滑开关。 取值范围：[0,1]。
xcorr_calc_upper_thd	相关计算上限阈值，值越大，计算得到的相位信息越多，但会增加计算资源。 取值范围：[-32,32]。
xcorr_calc_lower_thd	相关计算下限阈值，值越小，计算得到的相位信息越多，但会增加计算资源。 取值范围：[-32,32]。

成员名称	描述
luma_thd	前后帧平均亮度差异阈值，值越大，越容易进入因前后帧亮度变化引起的相位波动，不做对焦切换的策略。 取值范围：[0,128]。
phase_disparity_thd	相位差阈值，值越大，纳入统计的有效相位信息越多，但可能会引入异常相位。 取值范围：[0,32]。
dcc_slope_adj_coeff	dcc 斜率微调系数。 取值范围：[-1024,1024]。
confidence_thd_low	相位置信度下限阈值，小于该阈值，只用 CDAF 对焦。 取值范围：[-1024,1024]。
confidence_thd_high	相位置信度上限阈值，大于该阈值，只用 PDAF 对焦，介于两者之间，PDAF+CDAF 对焦。 取值范围：[-1024,1024]。
spc_en	shield pixel 亮度矫正使能开关。 取值范围：[0,1]。
left_spc_gain	left shield pixel 亮度增益系数。 取值范围：[0,1024]。
right_spc_gain	right shield pixel 亮度增益系数。 取值范围：[0,1024]。
dcc_slope	dcc 斜率系数。 取值范围：[-1024,1024]。
dcc_offset	dcc 偏置系数。 取值范围：[-1024,1024]。

【注意事项】

无

【相关数据类型及接口】

[xmedia_af_set_attr](#)

[xmedia_af_get_attr](#)

xmedia_af_cdaf_attr

【说明】

定义 cdaf 属性。

【定义】

```
typedef struct {  
} xmedia_af_cdaf_attr;
```

【成员】

成员名称	描述

【注意事项】

无

【相关数据类型及接口】

[xmedia_af_set_attr](#)

[xmedia_af_get_attr](#)

xmedia_af_info

【说明】

定义 af 状态信息。

【定义】

```
typedef struct {  
    xmedia_u64      frame_cnt;  
    xmedia_video_rect roi;  
  
    xmedia_u32 position;  
    xmedia_s32 defocus;  
    xmedia_s32 confidence_level;
```

```
xmedia_s32 phase_disparity;  
} xmedia_af_info;
```

【成员】

成员名称	描述
frame_cnt	当前帧计数。
roi	实际生效的 roi 区域。
position	镜头绝对位置。
defocus	镜头移动量。
confidence_level	PDAF 置信水平。
phase_disparity	PDAF 相位信息。

【注意事项】

无

【相关数据类型及接口】

[xmedia_af_query_info](#)

8 IMP

IMP 是指影响图像效果的模块，对应 API 接口必须在调用 `xmedia_isp_init` 接口之后才能调用。

8.1 HLC

8.1.1 功能描述

HLC (High Light Cover) 高光遮蔽模块，是用于将图像中亮度最高的区域（一般都是夜晚场景的车灯或者路灯等光源）拉灰，以减少类似于车灯等高亮光源对人眼的刺激。

8.1.2 API 参考

- `xmedia_isp_set_hlc_attr`: 设置高光遮蔽的属性。
- `xmedia_isp_get_hlc_attr`: 获取高光遮蔽的属性。

`xmedia_isp_set_hlc_attr`

【描述】

设置高光遮蔽的属性。

【语法】

```
xmedia_s32 xmedia_isp_set_hlc_attr(xmedia_u32 pipe, const xmedia_isp_hlc_attr *hlc_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
hlc_attr	图像的高光遮蔽属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

XM7206V11A 不支持该接口。

【举例】

无

【相关主题】

[xmedia_isp_get_hlc_attr](#)

xmedia_isp_get_hlc_attr

【描述】

获取图像的高光遮蔽属性。

【语法】

```
xmedia_s32 xmedia_isp_get_hlc_attr(xmedia_u32 pipe, xmedia_isp_hlc_attr *hlc_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
hlc_attr	图像的高光遮蔽属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

XM7206V11A 不支持该接口。

【举例】

无

【相关主题】

[xmedia_isp_set_hlc_attr](#)

8.1.3 数据类型

- [xmedia_isp_hlc_attr](#): 定义 hlc_attr 属性。

xmedia_isp_hlc_attr

【说明】

定义 hlc_attr 属性。

【定义】

```
typedef struct {
    xmedia_bool enable;
```

```

xmedia_u16 luma_thr;
xmedia_u16 luma_target;
} xmedia_isp_hlc_attr;

```

【成员】

成员名称	描述
enable	HLC 高光遮蔽的使能开关。 XMEDIA_FALSE：关闭； XMEDIA_TRUE：打开； 默认值：XMEDIA_FALSE；
luma_thr	高光遮蔽的阈值。亮度值大于该值的区域将会被拉灰成目标值为 luma_target 的低亮度值。 取值范围：[0, 1023]。默认值 940。 该值一般都设置的比较大，否则，图像的很多区域都会被拉灰。
luma_target	高光遮蔽拉灰的目标值。亮度值大于 luma_thr 的区域将被拉灰成目标值为 luma_target 的低亮度值，也即该区域的亮度将会被明显拉低拉暗。 取值范围：[0, 1023]。默认值 10。 该值一般都设置的比较小，以减少高亮光源对人眼的刺激。

8.2 Gamma

8.2.1 功能描述

Gamma 模块对图像进行亮度空间非线性转换以适配输出设备。Gamma 模块校正 R、G、B 通道时使用同一组 Gamma 表，Gamma 表各节点之间的间距相等，节点之间的图像像素值使用线性插值生成。

8.2.2 API 参考

- [xmedia_isp_set_gamma_attr](#)：设置 Gamma 属性。
- [xmedia_isp_get_gamma_attr](#)：获取 Gamma 属性。

[xmedia_isp_set_gamma_attr](#)

【描述】

设置 Gamma 属性。

【语法】

```
xmedia_s32 xmedia_isp_set_gamma_attr(xmedia_u32 pipe, const xmedia_isp_gamma_attr  
*gamma_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
gamma_attr	Gamma 属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无。

【举例】

无。

【相关主题】

[xmedia_isp_get_gamma_attr](#)

xmedia_isp_get_gamma_attr

【描述】

获取 Gamma 属性。

【语法】

```
xmedia_s32 xmedia_isp_get_gamma_attr(xmedia_u32 pipe, const xmedia_isp_gamma_attr  
*gamma_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
gamma_attr	Gamma 属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无。

【举例】

无。

【相关主题】

[xmedia_isp_set_gamma_attr](#)

8.2.3 数据类型

- [xmedia_isp_gamma_attr](#): 定义 ISP Gamma 校正属性。
- [xmedia_isp_gamma_curve_type](#): 定义 ISP Gamma 传输曲线类型。
- [xmedia_isp_gamma_manual_attr](#): 定义 gamma 手动属性。

- `xmedia_isp_gamma_auto_attr`: 定义 gamma 自动属性

xmedia_isp_gamma_attr

【说明】

定义 ISP Gamma 校正属性。

【定义】

```
typedef struct {
    xmedia_bool          enable;
    xmedia_isp_gamma_curve_type curve_type;
    xmedia_video_operation_mode op_type;
    xmedia_isp_gamma_manual_attr manual_attr;
    xmedia_isp_gamma_auto_attr auto_attr;
} xmedia_isp_gamma_attr;
```

说明

宏定义如下：

```
#define XMEDIA_ISP_ISO_MAX_COUNT    16
#define XMEDIA_ISP_GAMMA_NODE_NUM  1025
```

【成员】

成员名称	描述
enable	Gamma 校正功能使能。 XMEDIA_FALSE：关闭； XMEDIA_TRUE：使能。 默认值为 XMEDIA_TRUE。
curve_type	用来表示传输曲线类型，默认值为 XMEDIA_ISP_GAMMA_CURVE_TYPE_DEFAULT。
op_type	自动模式/手动模式选择。此功能在 curve_type 为 XMEDIA_ISP_GAMMA_CURVE_TYPE_USER_DEFINE 时生效。 0：自动 1：手动
manual_attr	手动属性。此功能在 curve_type 为 XMEDIA_ISP_GAMMA_CURVE_TYPE_USER_DEFINE 时生效。
auto_attr	自动属性。此功能在 curve_type 为

成员名称	描述
	XMEDIA_ISP_GAMMA_CURVE_TYPE_USER_DEFINE 时生效。

【注意事项】

- Gamma 校正 R、G、B 调用同一组 Gamma Table。
- curve_type 选择为 XMEDIA_ISP_GAMMA_CURVE_TYPE_DEFAULT、XMEDIA_ISP_GAMMA_CURVE_TYPE_SRGB、XMEDIA_ISP_GAMMA_CURVE_TYPE_HDR 等三种模式时，系统会自动配置预设的曲线，此时通过工具拖拉曲线均不生效。
- 只有选择 XMEDIA_ISP_GAMMA_CURVE_TYPE_USER_DEFINE 时，才能通过工具进行曲线修改和自动手动模式选择。
- 使用任意类型的预设曲线后，可以切换成 XMEDIA_ISP_GAMMA_CURVE_TYPE_USER_DEFINE 并以预设曲线为基础进行 Gamma 调试，**此时所有 auto 和 manual 下对应 table 均被重置为预设曲线（注意：要求切换到 XMEDIA_ISP_GAMMA_CURVE_TYPE_DEFAULT、XMEDIA_ISP_GAMMA_CURVE_TYPE_SRGB、XMEDIA_ISP_GAMMA_CURVE_TYPE_HDR 时手动点击界面 read 按钮，才可将 auto 和 manual 下对应 table 刷新为预设曲线）**，进而可以切到 XMEDIA_ISP_GAMMA_CURVE_TYPE_USER_DEFINE 模式以预设曲线开始调节。

【相关数据类型及接口】

无。

xmedia_isp_gamma_curve_type

【说明】

定义 Gamma 曲线类型。

【定义】

```
typedef enum {
    XMEDIA_ISP_GAMMA_CURVE_TYPE_DEFAULT = 0x0,
    XMEDIA_ISP_GAMMA_CURVE_TYPE_SRGB,
    XMEDIA_ISP_GAMMA_CURVE_TYPE_HDR,
    XMEDIA_ISP_GAMMA_CURVE_TYPE_USER_DEFINE,
```

```

XMEDIA_ISP_GAMMA_CURVE_TYPE_MAX
} xmedia_isp_gamma_curve_type;

```

【成员】

成员名称	描述
XMEDIA_ISP_GAMMA_CURVE_TYPE_DEFAULT	默认传输曲线。
XMEDIA_ISP_GAMMA_CURVE_TYPE_SRGB	默认 SDR 传输曲线。
XMEDIA_ISP_GAMMA_CURVE_TYPE_HDR	默认 HDR 传输曲线。
XMEDIA_ISP_GAMMA_CURVE_TYPE_USER_DEFINE	用户自定义传输曲线。
XMEDIA_ISP_GAMMA_CURVE_TYPE_MAX	传输曲线类型枚举终止位指示。

【注意事项】

用户自定义 Gamma 曲线时，必须确保 Gamma 表配置正确。相同预定义类型曲线在 WDR 和线性模式下实际 Gamma 曲线也不相同

【相关数据类型及接口】

无

xmedia_isp_gamma_manual_attr

【说明】

定义 Gamma 曲线手动属性。

【定义】

```

typedef enum {
    xmedia_u16 table[XMEDIA_ISP_GAMMA_NODE_NUM];
} xmedia_isp_gamma_manual_attr;

```

【成员】

成员名称	描述
table	手动 gamma 表。取值范围[0,4095]

xmedia_isp_gamma_auto_attr

【说明】

定义 Gamma 曲线自动属性。

【定义】

```
typedef enum {
    xmedia_u8          iso_num;
    xmedia_u32         iso_level[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16
    table[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_GAMMA_NODE_NUM];
} xmedia_isp_gamma_auto_attr;
```

【成员】

成员名称	描述
iso_num	iso 节点数量。
iso_level	iso level 等级。
table	各 iso 下 table 表设置。取值范围[0,4095]

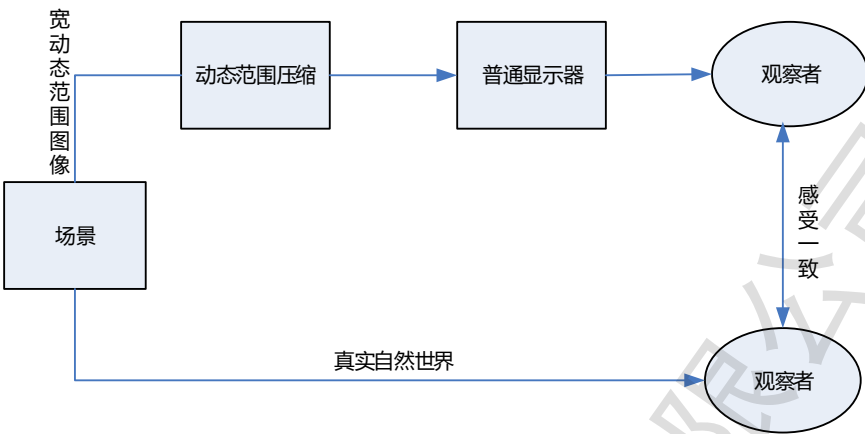
8.3 DRC

8.3.1 功能描述

DRC 算法用于对 WDR 合成后的数据进行动态范围压缩 (Dynamic Range Compression)。图像一般需要在显示设备上显示，CRT 显示器的动态范围一般只有 10²dB，而 WDR 合成后的数据的动态范围可以达到 10⁹dB，如果直接在 CRT 显示器上显示，就会由于动态范围不匹配的问题，造成丢失亮度较高或者较低处的细节。

DRC 算法的目的就是要使真实场景的观察者和显示设备的观察者都能获得相同的视觉感受。DRC 算法将高动态范围的图像压缩到显示器的动态范围，同时尽可能的保留原图像的细节和对比度。

图8-1 动态范围压缩



8.3.2 API 参考

- `xmedia_isp_set_drc_attr`: 设置动态范围压缩参数。
- `xmedia_isp_get_drc_attr`: 获取动态范围压缩参数。

`xmedia_isp_set_drc_attr`

【描述】

设置动态范围压缩参数。

【语法】

```
xmedia_s32 xmedia_isp_set_drc_attr(xmedia_u32 pipe, const xmedia_isp_drc_attr *drc_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
drc_attr	动态范围压缩参数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

DRC 使能时, 最小分辨率为 128x128。

【举例】

无。

【相关主题】

[xmedia_isp_get_drc_attr](#)

xmedia_isp_get_drc_attr

【描述】

获取动态范围压缩参数。

【语法】

```
xmedia_s32 xmedia_isp_get_drc_attr(xmedia_u32 pipe, xmedia_isp_drc_attr *drc_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
drc_attr	动态范围压缩参数。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败, 其值为错误码。

【需求】

- 头文件: xmedia_isp.h

- 库文件: libxmedia_isp.a

【注意】

无。

【举例】

无。

【相关主题】

[xmedia_isp_set_drc_attr](#)

8.3.3 数据类型

- [xmedia_video_operation_mode](#): 定义手动自动模式选项。
- [xmedia_isp_asymmetry_curve_attr](#): 定义 asymmetry 类型 TM 曲线
- [xmedia_isp_drc_tm_type](#): 定义 ISP DRC 曲线类型。
- [xmedia_isp_drc_manual_attr](#): 定义 isp_drc 手动属性。
- [xmedia_isp_drc_auto_attr](#): 定义 isp_drc 自动属性。
- [xmedia_isp_drc_attr](#): 定义 isp_drc 属性。

xmedia_video_operation_mode

【说明】

定义手动自动模式选项。

【定义】

```
typedef enum {  
    XMEDIA_VIDEO_OPERATION_MODE_AUTO    = 0,  
    XMEDIA_VIDEO_OPERATION_MODE_MANUAL = 1,  
    XMEDIA_VIDEO_OPERATION_MODE_MAX  
} xmedia_video_operation_mode;
```

【成员】

成员名称	描述
XMEDIA_VIDEO_OPERATION_MODE_AUTO	自动模式。
XMEDIA_VIDEO_OPERATION_MODE_MANUAL	手动模式。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_asymmetry_curve_attr

【说明】

定义 asymmetry 类型 TM 曲线。

【定义】

```
typedef struct {
    xmedia_u8 asymmetry;
    xmedia_u8 second_pole;
    xmedia_u8 stretch;
    xmedia_u8 compress;
} xmedia_isp_asymmetry_curve_attr;
```

【成员】

成员名称	描述
asymmetry	Asymmetry 曲线参数，取值范围：[0x1, 0x1E]。该值越小，暗区拉伸的程度越大
second_pole	Asymmetry 曲线参数，取值范围：[0x96, 0xD2]。该值越大，图像整体拉伸越大
stretch	Asymmetry 曲线参数，取值范围：[0x1E,0x3C]。该值越小，图像整体越亮
compress	Asymmetry 曲线参数，取值范围：[0x64,0xC8]。该值越大，在垂直方向曲线压缩程度越大，图像整体亮度越低

【注意事项】

Asymmetry 曲线的四个参数控制曲线在暗区及亮区亮度映射关系，详细使用说明可见《ISP 图像调优指南》

【相关数据类型及接口】

[xmedia_isp_drc_attr](#)

xmedia_isp_drc_tm_type

【说明】

定义 DRC 曲线类型。

【定义】

```
typedef enum {  
    XMEDIA_ISP_DRC_TM_TYPE_ASYMMETRY,  
    XMEDIA_ISP_DRC_TM_TYPE_USER_DEFINE,  
    XMEDIA_ISP_DRC_TM_TYPE_MAX  
} xmedia_isp_drc_tm_type;
```

【成员】

成员名称	描述
XMEDIA_ISP_DRC_TM_TYPE_ASYMMETRY	选择 Asymmetry Curve。
XMEDIA_ISP_DRC_TM_TYPE_USER_DEFINE	选择用户自定义曲线。
XMEDIA_ISP_DRC_TM_TYPE_MAX	无效值。

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_drc_attr](#)

xmedia_isp_drc_manual_attr

【说明】

定义 isp_drc 手动属性。

【定义】

```
typedef struct {  
    xmedia_u16 range_fil_coef;  
    xmedia_u16 spatial_fil_coef;  
    xmedia_u8  dering_thd;  
    xmedia_u8  dering_coarse_str;  
    xmedia_u16 dering_fine_str;  
    xmedia_u8  mixing_bright_min;  
    xmedia_u8  mixing_bright_max;  
    xmedia_u16 mixing_bright_thd_low;
```

```

xmedia_u16 mixing_bright_thd_high;
xmedia_u8  mixing_dark_min;
xmedia_u8  mixing_dark_max;
xmedia_u16 mixing_dark_thd_low;
xmedia_u16 mixing_dark_thd_high;
xmedia_u8  ltmc_str;
xmedia_u16 drc_str;
xmedia_isp_asymmetry_curve_attr asymmetry_curve;
xmedia_isp_asymmetry_curve_attr asymmetry_max_curve;
xmedia_u16 tm_value_user[XMEDIA_ISP_DRC_TM_VALUE_NUM];
xmedia_u8  tm_value_max_user[XMEDIA_ISP_DRC_TM_VALUE_NUM];
} xmedia_isp_drc_manual_attr;

```



说明

宏定义如下：

```
#define XMEDIA_ISP_DRC_TM_VALUE_NUM 1024
```

【成员】

成员名称	描述	取值范围	默认值
range_fil_coef	值域滤波系数。	XM7606V13 取值范围：[0x2, 0x7FFF] XM7206V11A 取值范围：[0x0, 0x400]	XM7606V13: Linear:3000 WDR:5980 XM7206V11A: 512
spatial_fil_coef	空域滤波系数。	XM7606V13 取值范围：[0x10, 0xFFFF] XM7206V11A 取值范围：[0x0, 0x400]	XM7606V13: 14463 XM7206V11A: 512
dering_thd	值域系数自适应强度值。	[0x0, 0xFF]	16
dering_coarse_str	值域系数自适应高强度值, XM7606V13 不支持。	[0x0, 0xFF]	16

dering_fine_str	值域系数自适应低亮度阈值。	[0x1, 0x3FF]	102
mixing_bright_min	正向细节增强低亮区域强度值。	[0x0, 0xFF]	128
mixing_bright_max	正向细节增强高亮区域强度值。	[0x0, 0xFF]	192
mixing_bright_thd_low	正向细节增强低亮度阈值。	[0x0, 0xFFF]	64
mixing_bright_thd_high	正向细节增强高亮度阈值。	[0x4, 0xFFF]	512
mixing_dark_min	负向细节增强低亮区域强度值。	[0x0, 0xFF]	128
mixing_dark_max	负向细节增强高亮区域强度值。	[0x0, 0xFF]	192
mixing_dark_thd_low	负向细节增强低亮度阈值。	[0x0, 0xFFF]	64
mixing_dark_thd_high	负向细节增强高亮度阈值。	[0x4, 0xFFF]	512
ltmc_str	local tomemapping 强度值。	[0x0, 0x20]	16
drc_str	drc 强度值; 强度越小, 画面亮度越接近 DRC 关闭	[0x0, 0x3FF]	256
asymmetry_curve	DRC ASYMMETRY 曲线相关参数。	asymmetry: [1, 30], second_pole:	(6, 180, 50, 150)

asymmetry_max_curve	LTMC ASYMMETRY 曲线相关参数。	[150, 210], stretch: [30, 60], compress: [100, 200];	
tm_value_user	用户自定义 DRC 曲线。	[0, 4095]	-
tm_value_max_user	用户自定义 LTMC 曲线。	[0, 255]	-

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_drc_attr](#)

xmedia_isp_drc_auto_attr

【说明】

定义 isp_drc 自动属性。

【定义】

```
typedef struct {
    xmedia_u8    iso_num;
    xmedia_u32   iso_level[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16 range_fil_coef[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16 spatial_fil_coef[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8    dering_thd[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8    dering_coarse_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16 dering_fine_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8    mixing_bright_min[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8    mixing_bright_max[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16 mixing_bright_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16 mixing_bright_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8    mixing_dark_min[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8    mixing_dark_max[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16 mixing_dark_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16 mixing_dark_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
}
```

```

xmedia_u8 ltmc_str[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 drc_str[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8 ratio_num;
xmedia_u16 ratio_level[XMEDIA_ISP_EXP_RATIO_MAX_COUNT];
xmedia_isp_asymmetry_curve_attr
asymmetry_curve[XMEDIA_ISP_EXP_RATIO_MAX_COUNT];
xmedia_isp_asymmetry_curve_attr asymmetry_max_curve[XMEDIA_ISP_
EXP_RATIO_MAX_COUNT];
xmedia_u16 tm_value_user[XMEDIA_ISP_EXP_RATIO_MAX_COUNT]
[XMEDIA_ISP_DRC_TM_VALUE_NUM];
xmedia_u8 tm_value_max_user[XMEDIA_ISP_EXP_RATIO_MAX_COUNT]
[XMEDIA_ISP_DRC_TM_VALUE_NUM];
} xmedia_isp_drc_auto_attr;

```

说明

宏定义如下：

```

#define XMEDIA_ISP_ISO_MAX_COUNT 16
#define XMEDIA_ISP_EXP_RATIO_MAX_COUNT 9

```

【成员】

成员名称	描述	取值范围	默认值
iso_num	ISO 节点数量。	[0x1, 0x10]	1
iso_level	ISO 节点。	[0x64, 0x320000]	100
range_fil_coef	值域滤波系数。	XM7606V13 取 值范围：[0x2, 0x7FFF] XM7206V11A 取 值范围：[0x0, 0x400]	XM7606V13: Linear:3000 WDR:5980 XM7206V11A: 512
spatial_fil_coef	空域滤波系数。	空域滤波系数。 XM7606V13 取 值范围：[0x10, 0xFFFF] XM7206V11A 取 值范围：[0x0, 0x400]	XM7606V13: 14463 XM7206V11A: 512

dering_thd	值域系数自适应强度值。	[0x0, 0xFF]	16
dering_coarse_str	值域系数自适应高强度值, XM7606V13 不支持。	[0x0, 0xFF]	16
dering_fine_str	值域系数自适应低亮度阈值。	[0x0, 0x3FF]	102
mixing_bright_min	正向细节增强低亮区域强度值。	[0x0, 0xFF]	128
mixing_bright_max	正向细节增强高亮区域强度值。	[0x0, 0xFF]	192
mixing_bright_thd_low	正向细节增强低亮度阈值。	[0x0, 0xFFFF]	64
mixing_bright_thd_high	正向细节增强高亮度阈值。	[0x4, 0xFFFF]	512
mixing_dark_min	负向细节增强低亮区域强度值。	[0x0, 0xFF]	128
mixing_dark_max	负向细节增强高亮区域强度值。	[0x0, 0xFF]	192
mixing_dark_thd_low	负向细节增强低亮度阈值。	[0x0, 0xFFFF]	64
mixing_dark_thd_high	负向细节增强高亮度阈值。	[0x4, 0xFFFF]	512
ltmc_str	local tomemapping 强度值。	[0x0, 0x20]	16

drc_str	drc 强度值。	[0x0, 0x3FF]	256
ratio_num	曝光比节点数量。	[0x1, 0x9]	1
ratio_level	曝光比节点。	[0x40, 0x4000]	64
asymmetry_curve	DRC ASYMMETRY 曲线相关参数。	asymmetry: [1, 30], second_pole: [150, 210], stretch: [30, 60], compress: [100, 200];	(6, 180, 50, 150)
asymmetry_max_curve	LTMC ASYMMETRY 曲线相关参数。		
tm_value_user	用户自定义 DRC 曲线。	[0, 4095]	-
tm_value_max_user	用户自定义 LTMC 曲线。	[0, 256]	-

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_drc_attr](#)

xmedia_isp_drc_attr

【说明】

定义 ISP DRC 属性。

【定义】

```
typedef struct {
    xmedia_bool    drc_en;
    xmedia_bool    ltmc_en;
    xmedia_bool    dering_en;
    xmedia_bool    detail_enhance_en;
    xmedia\_isp\_drc\_tm\_type tm_type;
}
```

```

xmedia_video_operation_mode op_type;
xmedia_isp_drc_manual_attr   manual_attr;
xmedia_isp_drc_auto_attr     auto_attr;
} xmedia_isp_drc_attr;

```

【成员】

成员名称	描述
drc_en	DRC 使能开关。
ltmc_en	LTMC 开关。
dering_en	Dering 功能开关
detail_enhance_en	细节增强开关。
tm_type	DRC 曲线选择。
op_type	DRC 操作模式。
manual_attr	DRC 手动模式。
auto_attr	DRC 自动模式。

【注意事项】

无

【相关数据类型及接口】

无

8.4 Mesh Shading

8.4.1 功能描述

LSC 模块主要用来处理由于镜头光学折射不均匀导致的镜头周围出现阴影的情况。目前流行的处理方式有 Radial（同轴圆）方式以及 Mesh（网格）方式，本模块采用 Mesh（网格）方式。

- 算法会将整个 Bayer 域画面分割成 32*32 个子区域，这 32*32 个区域大小大致相等。

在数据处理过程中，算法将分别对 RAW 域中的每个通道进行处理。

8.4.2 API 参考

- [xmedia_esp_set_mesh_shading_attr](#): 设定 Mesh Shading 算法参数。
- [xmedia_esp_get_mesh_shading_attr](#): 获取 Mesh Shading 算法参数。
- [xmedia_esp_set_mesh_shading_gain_lut_attr](#): 设定 Mesh Shading 增益表属性。
- [xmedia_esp_get_mesh_shading_gain_lut_attr](#): 获取 Mesh Shading 增益表属性。

xmedia_esp_set_mesh_shading_attr

【描述】

设置 Mesh Shading 算法参数。

【语法】

```
xmedia_s32 xmedia_esp_set_mesh_shading_attr(xmedia_u32 pipe, const
xmedia_esp_mesh_shading_attr *shading_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
shading_attr	Mesh Shading 算法参数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_esp.h
- 库文件: libxmedia_esp.a

【注意】

无。

【举例】

无。

【相关主题】

[xmedia_isp_get_mesh_shading_attr](#)

xmedia_isp_get_mesh_shading_attr

【描述】

获取 Mesh Shading 算法参数。

【语法】

```
xmedia_s32 xmedia_isp_get_mesh_shading_attr(xmedia_u32 pipe,  
xmedia\_isp\_mesh\_shading\_attr *shading_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
shading_attr	shading_attr 算法参数。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_isp.h
- 库文件：libxmedia_isp.a

【注意】

无。

【举例】

无。

【相关主题】

[xmedia_isp_set_mesh_shading_attr](#)

xmedia_isp_set_mesh_shading_gain_lut_attr

【描述】

设定 Mesh Shading 增益表属性。

【语法】

```
xmedia_s32 xmedia_isp_set_mesh_shading_gain_lut_attr(xmedia_u32 pipe, const  
xmedia\_isp\_shading\_lut\_attr *shading_gain_lut_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
shading_gain_lut_attr	Mesh Shading 增益表属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无。

【举例】

无。

【相关主题】

[xmedia_isp_get_mesh_shading_gain_lut_attr](#)

xmedia_isp_get_mesh_shading_gain_lut_attr

【描述】

获取 Mesh Shading 增益表属性。

【语法】

```
xmedia_s32 xmedia_isp_get_mesh_shading_gain_lut_attr(xmedia_u32 pipe, const  
xmedia\_isp\_shading\_lut\_attr *shading_gain_lut_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
shading_gain_lut_attr	Mesh Shading 增益表属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_isp.h
- 库文件：libxmedia_isp.a

【注意】

无。

【举例】

无。

【相关主题】

[xmedia_isp_set_mesh_shading_gain_lut_attr](#)

8.4.3 数据类型

- [xmedia_isp_mesh_shading_attr](#): 定义 Mesh Shading 算法参数。
- [xmedia_isp_shading_gain_lut](#): 定义 Mesh Shading 增益表参数。
- [xmedia_isp_shading_lut_attr](#): 定义 Mesh Shading 增益表属性。

[xmedia_isp_mesh_shading_attr](#)

【说明】

定义 Mesh Shading 算法参数。

【定义】

```
typedef struct {  
    xmedia_bool enable;  
    xmedia_u8    iso_num;  
    xmedia_u32   iso_level[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u16   mesh_str[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u16   ct_num;  
    xmedia_u16   ct_level[XMEDIA_ISP_CT_MAX_COUNT];  
    xmedia_u16   blend_ratio[XMEDIA_ISP_CT_MAX_COUNT];  
} xmedia_isp_mesh_shading_attr;
```



说明

宏定义如下：

```
#define XMEDIA_ISP_ISO_MAX_COUNT    16  
#define XMEDIA_ISP_CT_MAX_COUNT    16
```

【成员】

成员名称	描述	取值范围	默认值
enable	Mesh Shading 算法开关。	[0,1] 0：禁止； 1：使能；	0
iso_num	ISO 数量。	[1, 16]	
iso_level	ISO 等级。	[100, 3276800]	
mesh_str	用来全局纠正 Mesh Shading 标定后的强度。当镜头 shading 很严重的时候，	[0,256]	256

成员名称	描述	取值范围	默认值
	画面四个角补偿的增益很大，容易导致噪声很大，而且画面有格子现象。这个调整 u16MeshStrength 小于一倍默认值，可以减少 Mesh Shading 的补偿值，让四个角的补偿增益较小一些，达到优化噪声和画面格子问题。		
ct_num	色温数量。	[1, 16]	
ct_level	色温等级。	[2000, 9500]	
blend_ratio	两张增益表的融合比例，表示为第 0 张表的权重值，256 表示只选择第 0 张表，0 表示只选择第 1 张表。	[0,256]	256

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_set_mesh_shading_attr](#)
[xmedia_isp_get_mesh_shading_attr](#)

xmedia_isp_shading_gain_lut

【说明】

定义 Mesh Shading 增益表参数。

【定义】

```
typedef struct {
    xmedia_u16 rgain[XMEDIA_ISP_LSC_GRID_POINTS];
    xmedia_u16 grgain[XMEDIA_ISP_LSC_GRID_POINTS];
    xmedia_u16 gbgain[XMEDIA_ISP_LSC_GRID_POINTS];
    xmedia_u16 bgain[XMEDIA_ISP_LSC_GRID_POINTS];
} xmedia_isp_shading_gain_lut;
```



说明

宏定义如下：

```
#define XMEDIA_ISP_LSC_GRID_POINTS    (33*33)
```

【成员】

成员名称	描述	取值范围	默认值
rgain	用来储存 Mesh Shading 所用 R 通道标定数据。该数组表示画面从左至右、从上至下的网格交点处的 R 分量阴影矫正增益数据值。	[0,1023]	128
grgain	用来储存 Mesh Shading 所用 GR 通道标定数据。该数组表示画面从左至右、从上至下的网格交点处的 GR 分量阴影矫正增益数据值。	[0,1023]	128
gbgain	用来储存 Mesh Shading 所用 GB 通道标定数据。该数组表示画面从左至右、从上至下的网格交点处的 GB 分量阴影矫正增益数据值。	[0,1023]	128
bgain	用来储存 Mesh Shading 所用 B 通道标定数据。该数组表示画面从左至右、从上至下的网格交点处的 B 分量阴影矫正增益数据值。	[0,1023]	128

【注意事项】

无

【相关数据类型及接口】

- [xmedia_esp_shading_lut_attr](#)
- [xmedia_esp_set_mesh_shading_gain_lut_attr](#)
- [xmedia_esp_get_mesh_shading_gain_lut_attr](#)

xmedia_esp_shading_lut_attr

【说明】

定义 Mesh Shading 增益表属性。

【定义】

```
typedef struct {
```

```

xmedia_u8          mesh_scale;
xmedia_u16         xgrid_width[(XMEDIA_ISP_LSC_GRID_COL - 1) / 2];
xmedia_u16         ygrid_width[(XMEDIA_ISP_LSC_GRID_ROW - 1) / 2];
xmedia_isp_shading_gain_lut lsc_gain_lut[XMEDIA_ISP_LSC_GAIN_LUT_NUM];
} xmedia_isp_shading_lut_attr;

```



说明

宏定义如下：

```

#define XMEDIA_ISP_LSC_GRID_COL      33
#define XMEDIA_ISP_LSC_GRID_ROW      33
#define XMEDIA_ISP_LSC_GAIN_LUT_NUM  2

```

【成员】

成员名称	描述	取值范围	默认值
mesh_scale	增益表控制精度	[0,7]	7
xgrid_width	用来储存各 GRID 分区宽度大小信息 暂未支持	[4,255]	
ygrid_width	用来储存各 GRID 分区高度大小信息 暂未支持	[4,255]	
lsc_gain_lut	两组色温下的增益表配置。硬件基于两组表以及 blend_ratio 进行当前色温下校正增益表的计算。		

【注意事项】

无

【相关数据类型及接口】

- [xmedia_isp_set_mesh_shading_gain_lut_attr](#)
- [xmedia_isp_get_mesh_shading_gain_lut_attr](#)

8.5 DPC

8.5.1 功能描述

DPC 算法通过在 5x5 的窗口中通过某些坏点检测算法找到该窗口中明显异于临近像素的坏点。该模块主要包含以下两种模式：

- 静态坏点标定/校正有两种流程：亮点和暗点流程

- 在亮点流程中，光圈处于关闭状态，启动坏点标定程序，得到坏点坐标信息。坏点个数的总数由坏点校正模块的 memory 决定的。得到的坏点通过临近像素的中值滤波进行校正。
- 在暗点流程中，光圈处于正常打开状态，要求平坦背景，最好使用灰度箱固定光源，图像整体平均亮度大约为最大亮度 50%，或 Bayer 格式中 B 通道亮度为最大亮度 20% 左右，启动坏点检测程序，得到坏点坐标。坏点个数的总数是由坏点校正模块的 memory 决定的。得到的坏点通过临近像素的中值滤波进行校正。

- 动态坏点检测/校正

在这种模式中，校正模块使用动态检测方法的进行坏点检测，能够校正的坏点个数是有限制的。这种模式总体上相对于静态模式更不可靠，但是在低照度情况下强烈推荐使用动态坏点校正功能，可以校正更多的坏点以及明显改善图像偏色的问题。

8.5.2 API 参考

- [xmedia_isp_set_dpc_calibrate](#)：设置静态坏点标定参数。
- [xmedia_isp_get_dpc_calibrate](#)：获取静态坏点标定参数。
- [xmedia_isp_set_dpc_static_attr](#)：设置静态坏点校正属性。
- [xmedia_isp_get_dpc_static_attr](#)：获取静态坏点校正属性。
- [xmedia_isp_set_dpc_dynamic_attr](#)：设置动态坏点校正属性。
- [xmedia_isp_get_dpc_dynamic_attr](#)：获取动态坏点校正属性。

xmedia_isp_set_dpc_calibrate

【描述】

设置静态坏点标定参数。

【语法】

```
xmedia_s32 xmedia_isp_set_dpc_calibrate(xmedia_u32 pipe, const
xmedia\_isp\_dpc\_static\_calibrate *dpc_calibrate);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
dpc_calibrate	静态坏点标定参数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

该接口为静态坏点标定接口，坏点标定只需启动一次，标定完成之后自动关闭标定使能，恢复为正常的坏点校正模式。亮坏点检测的环境为：使用最小的模拟增益和数字增益，降帧率到 5 ~ 6fps，使曝光时间为 200ms（系统自动完成），遮黑镜头或关闭光圈（需手动操作）；暗坏点检测的环境为：光圈处于正常打开状态，要求平坦背景，最好使用辉度箱固定光源，图像整体平均亮度大约为最大亮度 50%，或 Bayer 格式中 B 通道亮度为最大亮度 20%左右。

非 PIPE1 通路，该标定功能，暂未支持。

XM7206V11A 不支持静态坏点标定。

【相关主题】

[xmedia_isp_get_dpc_calibrate](#)

xmedia_isp_get_dpc_calibrate

【描述】

获取静态坏点标定结果。

【语法】

```
xmedia_s32 xmedia_isp_get_dpc_calibrate(xmedia_u32 pipe, const  
xmedia\_isp\_dpc\_static\_calibrate *dpc_calibrate);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
dpc_calibrate	静态坏点标定参数。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

XM7206V11A 不支持静态坏点标定。

【相关主题】

[xmedia_isp_set_dpc_calibrate](#)

xmedia_isp_set_dpc_static_attr

【描述】

设置静态坏点属性，合并坏点表，将坏点坐标信息与坏点数目写入系统内存。

【语法】

```
xmedia_s32 xmedia_isp_set_dpc_static_attr(xmedia_u32 pipe, const xmedia\_isp\_dpc\_static\_attr  
*dpc_static_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
dpc_static_attr	静态坏点属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

XM7206V11A 不支持静态坏点矫正。

【相关主题】

[xmedia_isp_get_dpc_static_attr](#)

xmedia_isp_get_dpc_static_attr

【描述】

获取静态坏点属性。

【语法】

```
xmedia_s32 xmedia_isp_get_dpc_static_attr(xmedia_u32 pipe, xmedia_isp_dpc_static_attr
*dpc_static_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
dpc_static_attr	静态坏点属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

在设置静态坏点属性中，亮点坏点表中表示亮坏点坐标信息，暗坏点表中则保存的是暗坏点坐标信息；而获取静态坏点属性时，亮点坏点表中保存的是所有坏点的坐标信息。

XM7206V11A 不支持静态坏点矫正。

【相关主题】

[xmedia_isp_set_dpc_static_attr](#)

[xmedia_isp_set_dpc_dynamic_attr](#)

【描述】

设置动态坏点调试属性。

【语法】

```
xmedia_s32 xmedia_isp_set_dpc_dynamic_attr(xmedia_u32 pipe, const  
xmedia\_isp\_dpc\_dynamic\_attr *dpc_dynamic_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
dpc_dynamic_attr	动态坏点属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【相关主题】

[xmedia_isp_get_dpc_dynamic_attr](#)

xmedia_isp_get_dpc_dynamic_attr

【描述】

获取动态坏点调试属性。

【语法】

```
xmedia_s32 xmedia_isp_get_dpc_dynamic_attr(xmedia_u32 pipe,  
xmedia\_isp\_dpc\_dynamic\_attr *dpc_dynamic_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
dpc_dynamic_attr	动态坏点属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_isp.h
- 库文件：libxmedia_isp.a

【注意】

无。

【相关主题】

[xmedia_isp_set_dpc_dynamic_attr](#)

8.5.3 数据类型

- [xmedia_isp_static_dpc_type](#)：静态坏点标定类型。
- [xmedia_isp_dpc_status](#)：ISP 校正（检测）状态。
- [xmedia_isp_dpc_static_calibrate](#)：静态坏点标定属性。
- [xmedia_isp_dpc_static_attr](#)：静态坏点校正属性。
- [xmedia_isp_dpc_dynamic_manual_attr](#)：定义动态坏点校正属性。
- [xmedia_isp_dpc_dynamic_auto_attr](#)：定义动态坏点校正的手动属性。
- [xmedia_isp_dpc_dynamic_attr](#)：定义动态坏点校正的自动属性。

xmedia_isp_static_dpc_type

【说明】

定义静态坏点标定类型。

【定义】

```
typedef enum {
    XMEDIA_ISP_STATIC_DPC_TYPE_BRIGHT = 0X0,
    XMEDIA_ISP_STATIC_DPC_TYPE_DARK,
    XMEDIA_ISP_STATIC_DPC_TYPE_MAX
} xmedia_isp_static_dpc_type;
```

【成员】

成员名称	描述
XMEDIA_ISP_STATIC_DPC_TYPE_BRIGHT	0x0: 选择亮点标定。
XMEDIA_ISP_STATIC_DPC_TYPE_DARK	0x1: 选择暗点标定。

【注意事项】

XM7206V11A 不支持静态坏点标定。

【相关数据类型及接口】

[xmedia_isp_dpc_static_calibrate](#)

[xmedia_isp_dpc_status](#)

【说明】

定义 ISP 校正（检测）状态。

【定义】

```
typedef enum {  
    XMEDIA_ISP_DPC_STATUS_INIT      = 0,  
    XMEDIA_ISP_DPC_STATUS_SUCCESS = 1,  
    XMEDIA_ISP_DPC_STATUS_TIMEOUT = 2,  
    XMEDIA_ISP_DPC_STATUS_MAX  
} xmedia_isp_dpc_status;
```

【成员】

成员名称	描述
XMEDIA_ISP_DPC_STATUS_INIT	0: 初始状态，未标定。
XMEDIA_ISP_DPC_STATUS_SUCCESS	1: 静态坏点标定成功结束。
XMEDIA_ISP_DPC_STATUS_TIMEOUT	2: 静态坏点标定超时结束。

【注意事项】

只读。**XM7206V11A 不支持静态坏点标定。**

【相关数据类型及接口】

[xmedia_isp_dpc_static_calibrate](#)

xmedia_isp_dpc_static_calibrate

【说明】

定义静态坏点标定的属性。

【定义】

```
typedef struct {  
    xmedia_bool          enable_detect;  
    xmedia\_isp\_static\_dpc\_type static_dpc_type;  
    xmedia_u8            start_thd;  
    xmedia_u16           count_max;  
    xmedia_u16           count_min;  
    xmedia_u16           time_limit;  
    xmedia_u32           table[XMEDIA_ISP_STATIC_DPC_COUNT_MAX];  
    xmedia_u8            finish_thd;  
    xmedia_u16           count;  
    xmedia\_isp\_dpc\_status status;  
} xmedia_isp_dpc_static_calibrate;
```



说明

```
#define XMEDIA_ISP_STATIC_DPC_COUNT_MAX    4096
```

【成员】

成员名称	描述	取值范围	默认值
enable_detect	静态坏点标定使能。	[0,1] 0: 禁止 1: 使能	0
static_dpc_type	静态坏点类型。	[0,1] 0: 亮点 1: 暗点	0
start_thd	静态坏点标定开始时的检测门限值，该值的设置与 sensor 相关。	[1,0xFF]	0x3
count_max	允许静态坏点的最大个数。	(0,XMEDIA_ISP_STATIC_DPC_COUNT_MAX)	0x800

成员名称	描述	取值范围	默认值
		UNT_MAX]	
count_min	允许静态坏点的最小个数。	[0,count_max]	0x1
time_limit	允许标定超时门限值。	[0,0x640]	0x640
table	只读，亮暗坏点坐标查找表，低 29bit 有效，[12:0]bit 为坏点的水平坐标，[28:16]bit 为坏点的垂直坐标。	每个坐标取值范围 [0,0x1FFF1FFF]	0x0
finish_thd	只读，静态坏点标定结束时的检测门限值。	[0,0xFF]	0x3
count	只读，标定出的静态坏点的个数。	[0,XMEDIA_ISP_STATIC_DPC_COUNT_MAX]	0
status	只读，静态坏点标定结果状态信息。	[0,2]	0

【注意事项】

XM7206V11A 不支持静态坏点标定。

【相关数据类型及接口】

- [xmedia_isp_static_dpc_type](#)
- [xmedia_isp_dpc_status](#)

xmedia_isp_dpc_static_attr

【说明】

定义静态坏点校正属性。

【定义】

```
typedef struct {
    xmedia_bool  enable;
    xmedia_bool  pd_enable;
    xmedia_u16   bright_count;
    xmedia_u16   dark_count;
    xmedia_u32   bright_table[XMEDIA_ISP_STATIC_DPC_COUNT_MAX];
}
```

```

xmedia_u32 dark_table[XMEDIA_ISP_STATIC_DPC_COUNT_MAX];
xmedia_bool show;
} xmedia_isp_dpc_static_attr;

```

【成员】

成员名称	描述	取值范围	默认值
enable	静态坏点校正使能。	[0,1] 0: 禁止; 1: 使能;	1
pd_enable	pdaf 点的校正使能。	[0,1] 0: 禁止; 1: 使能;	0
bright_count	亮坏点个数。 注意: 当该变量作为输入时, 表示亮坏点的个数; 作为输出时, 表示总的坏点个数。	[0,XMEDIA_ISP_STATIC_DPC_COUNT_MAX]	0
dark_count	暗坏点个数。 注意: 当该变量作为输入时, 表示暗坏点的个数; 作为输出时, 无效值为 0;	[0,XMEDIA_ISP_STATIC_DPC_COUNT_MAX]	0
bright_table	亮坏点坐标信息, 低 29bit 有效, [12:0]bit 为坏点水平坐标, [28:16]为坏点垂直坐标。 注意: 当该变量作为输入时, 表示亮坏点的坐标值查找表; 作为输出时, 表示所有坏点的坐标值。	每个坐标的取值范围都是 [0,0x1FFF1FFF]	0
dark_table	暗的坏点坐标值, 低 29bit 有效, [12:0]bit 坏点水平坐标, [28:16]为坏点垂直坐标。 注意: 当该变量作为输入时, 表示暗坏点的坐标值查找表; 作为输出时, 无效值。	每个坐标的取值范围都是 [0,0x1FFF1FFF]	0

成员名称	描述	取值范围	默认值
show	静态坏点显示使能。	[0,1] 0: 禁止; 1: 使能;	0

【注意事项】

XM7206V11A 不支持静态坏点标定。

【相关数据类型及接口】

无。

xmedia_isp_dpc_dynamic_manual_attr

【说明】

定义动态坏点校正的手动属性。

【定义】

```
typedef struct {
    xmedia_u16 str;
    xmedia_u16 thd;
    xmedia_u16 slope;
    xmedia_u16 bright_thd;
    xmedia_u8 bright_bias;
    xmedia_u16 dark_thd;
    xmedia_u8 dark_bias;
} xmedia_isp_dpc_dynamic_manual_attr;
```

【成员】

成员名称	描述	取值范围	默认值
str	动态坏点检测强度，取值越大，检测到的坏点就越多。	XM7606V13: [1,120] XM7206V11A: [1,640]	1
thd	校正值与原像素融合阈值，数值越大，原像素占比越大，数值越小，校正值占比越大。	[63,4095]	2048

成员名称	描述	取值范围	默认值
slope	斜率，与 thd 搭配使用，数值越小，坡度越陡峭，数值越大，坡度越平缓。	[0,7]	1
bright_thd	亮区阈值，大于阈值的像素范围，可以通过 bright_bias 调节亮区坏点检测强度； XM7606V13 暂未支持；	[128,4095]	4000
bright_bias	亮区坏点检测强度调整参数，值越大，检测强度越强。6bit 小数精度； XM7606V13 暂未支持；	[0,127]	0
dark_thd	暗区阈值，小于阈值的像素范围，可以通过 dark_bias 调节暗区坏点检测强度； XM7606V13 暂未支持；	[0,3967]	200
dark_bias	暗区坏点检测强度调整参数，值越大，检测强度越强。6bit 小数精度； XM7606V13 暂未支持；	[0,127]	127

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_dpc_dynamic_attr](#)

[xmedia_isp_dpc_dynamic_auto_attr](#)

【说明】

定义动态坏点校正的自动属性。

【定义】

```
typedef struct {
    xmedia_u8  iso_num;
    xmedia_u32 iso_level[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16 str [XMEDIA_ISP_ISO_MAX_COUNT];
}
```

```

xmedia_u16 thd[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 slope[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 bright_thd[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8 bright_bias[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 dark_thd[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8 dark_bias[XMEDIA_ISP_ISO_MAX_COUNT];
} xmedia_isp_dpc_dynamic_auto_attr;

```



说明

```
#define XMEDIA_ISP_ISO_MAX_COUNT 16
```

【成员】

成员名称	描述	取值范围
iso_num	ISO 数目。	
iso_level	ISO 等级。	
str	动态坏点检测强度，值越大，检测到的坏点就越多。	XM7606V13: [1,120] XM7206V11A: [1,640]
thd	校正值与原像素融合阈值，数值越大，原像素占比越大，数值越小，校正值占比越大。	[63,4095]
slope	斜率，与 thd 搭配使用，数值越小，坡度越陡峭，数值越大，坡度越平缓；	[0,7]
bright_thd	亮区阈值，大于阈值的像素范围，可以通过 bright_bias 调节亮区坏点检测强度； XM7606V13 暂未支持；	[128,4095]
bright_bias	亮区坏点检测强度调整参数，值越大，检测强度越强。6bit 小数精度； XM7606V13 暂未支持；	[0,127]
dark_thd	暗区阈值，小于阈值的像素范围，可以通过 dark_bias	[0,3967]

成员名称	描述	取值范围
	调节暗区坏点检测强度; XM7606V13 暂未支持;	
dark_bias	暗区坏点检测强度调整参数, 值越大, 检测强度越强。6bit 小数精度; XM7606V13 暂未支持;	[0,127]

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_dpc_dynamic_attr](#)

xmedia_isp_dpc_dynamic_attr

【说明】

定义动态坏点校正属性。

【定义】

```
typedef struct {
    xmedia_bool          enable;
    xmedia_bool          sup_twinkle_en;
    xmedia_s8           soft_thd;
    xmedia_u8           soft_slope;
    xmedia_video_operation_mode op_type;
    xmedia_isp_dpc_dynamic_manual_attr manual_attr;
    xmedia_isp_dpc_dynamic_auto_attr auto_attr;
} xmedia_isp_dpc_dynamic_attr;
```

【成员】

成员名称	描述	取值范围	默认值
enable	动态坏点校正功能使能。	[0,1] 0: 禁止; 1: 使能;	0
sup_twinkle_en	闪烁抑制使能。	[0,1]	0

成员名称	描述	取值范围	默认值
	XM7606V13、XM7206V11A 暂未支持;	0: 禁止; 1: 使能;	
soft_thd	闪烁抑制的阈值, 值越小, 所能处理的像素点越多, 图像会越模糊。 XM7606V13、XM7206V11A 暂未支持;	[0,127]	6
soft_slope	闪烁抑制模块中, 控制偏差小于 soft_thr 的像素点的校正情况。值越大, 参与校正处理的像素点越多, 图像会越模糊。 XM7606V13、XM7206V11A 暂未支持;	[0,255]	21
op_type	动态坏点校正操作模式。	[0,1] 0: 自动; 1: 手动;	0
manual_attr	动态坏点校正手动模式。		
auto_attr	动态坏点校正自动模式。		

【注意事项】

无

【相关数据类型及接口】

- [xmedia_video_operation_mode](#)
- [xmedia_isp_dpc_dynamic_manual_attr](#)
- [xmedia_isp_dpc_dynamic_auto_attr](#)

8.6 Crosstalk

8.6.1 概述

Crosstalk Removal 模块的主要功能是为了平衡 rawdata 之间临近像素 Gr 和 Gb 之间的差异，能够有效防止 demosaic 插值算法产生的方格或其他类似 pattern。由于 sensor 可能会因为特殊角度的光线入射而产生 Crosstalk，形成一些 pattern，根本原因就是临近像素值之间 Gr 和 Gb 值域不一致。

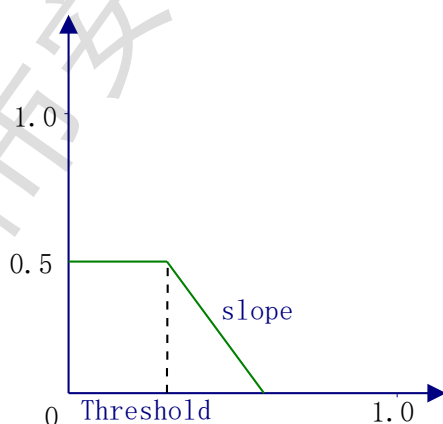
8.6.2 功能描述

crosstalk 模块的主要功能是为了平衡 rawdata 之间临近像素 Gr 和 Gb 之间的差异，能够有效防止 demosaic 插值算法产生的方格或其他类似 pattern。由于 sensor 可能会因为特殊角度的光线入射而产生 crosstalk，形成这些 pattern 的根本原因就是临近像素值之间 Gr 和 Gb 值不一致。

如图 8-2 所示，横坐标表示 Gr 与 Gb 之间的差值，即 $|Gr-Gb|$ ，纵坐标表示处理的强度值。图中的 Threshold，slope 与参数中的 au16Threshold，u8Slope 不完全对应，只是算法的大致示意。

- 当 Gr 与 Gb 之间的差值小于 threshold 值时，都按照最大的强度值 0.5 进行处理，Threshold 值越大，图像整体被处理的强度越大；
- 当 Gr 与 Gb 之间的差值大于 threshold 值时，由 slope 控制处理的强度，当 slope 值越大，整体处理强度衰减得越慢，也就是处理的强度越大。

图8-2 Crosstalk 门限



8.6.3 API 参考

- `xmedia_isp_set_crosstalk_attr`: 设定 crosstalk 属性。
- `xmedia_isp_get_crosstalk_attr`: 获取 crosstalk 属性。

`xmedia_isp_set_crosstalk_attr`

【描述】

设定 Crosstalk 属性。

【语法】

```
xmedia_s32 xmedia_isp_set_crosstalk_attr(xmedia_u32 pipe, const xmedia_isp_cr_attr *cr_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
cr_attr	Crosstalk 属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: `xmedia_isp.h`
- 库文件: `libxmedia_isp.a`

【注意】

无。

【举例】

无。

【相关主题】

[xmedia_isp_get_crosstalk_attr](#)

xmedia_isp_get_crosstalk_attr

【描述】

获取 Crosstalk 属性。

【语法】

```
xmedia_s32 xmedia_isp_get_crosstalk_attr(xmedia_u32 pipe, xmedia\_isp\_cr\_attr *cr_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
cr_attr	Crosstalk 属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无。

【举例】

无。

【相关主题】

[xmedia_isp_set_crosstalk_attr](#)

8.6.4 数据类型

- [xmedia_isp_cr_manual_attr](#): 定义 ISP Crosstalk 手动属性。
- [xmedia_isp_cr_auto_attr](#): 定义 ISP Crosstalk 自动属性。
- [xmedia_isp_cr_attr](#): 定义 ISP Crosstalk 属性。

xmedia_isp_cr_manual_attr

【说明】

定义 ISP crosstalk 手动属性。

【定义】

```
typedef struct {
    xmedia_u8  slope;
    xmedia_u16 threshold;
    xmedia_u16 np_offset;
} xmedia_isp_cr_manual_attr;
```

【成员】

成员名称	描述
slope	设置 crosstalk 门限之上的处理强度。值越大，表示在门限值 threshold 之上整体处理强度衰减得越慢。
threshold	设置 crosstalk 门限值。值越大，表示整体处理的强度越大。
np_offset	设置 noise profile 值。值越大，表示抗噪能力越大。 暂未支持

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_cr_attr](#)

xmedia_isp_cr_auto_attr

【说明】

定义 ISP Crosstalk 自动属性。

【定义】

```
typedef struct {
    xmedia_u8    iso_num;
    xmedia_u32 iso_level[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8    slope[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16 threshold[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16 np_offset[XMEDIA_ISP_ISO_MAX_COUNT];
} xmedia_isp_cr_auto_attr;
```

 说明
#define XMEDIA_ISP_ISO_MAX_COUNT 16

【成员】

成员名称	描述
iso_num	ISO 数量。
iso_level	ISO 等级。
slope	设置 Crosstalk 门限之上的处理强度。值越大，表示在门限值 threshold 之上整体处理强度衰减得越慢。
threshold	设置 Crosstalk 门限值。值越大，表示整体处理的强度越大。
np_offset	设置 Noise Profile 值。值越大，表示抗噪能力越大。 暂未支持

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_cr_attr](#)

xmedia_isp_cr_attr

【说明】

定义 ISP crosstalk 属性。

【定义】

```
typedef struct {
    xmedia_bool                enable;
    xmedia\_video\_operation\_mode op_type;
    xmedia\_isp\_cr\_manual\_attr manual_attr;
```

```

xmedia_isp_cr_auto_attr auto_attr;
} xmedia_isp_cr_attr;

```

【成员】

成员名称	描述
enable	使能 crosstalk 功能。
op_type	CR 操作模式。
manual_attr	CR 手动模式。
auto_attr	CR 自动模式。

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_set_crosstalk_attr](#)

[xmedia_isp_get_crosstalk_attr](#)

8.7 去噪算法

8.7.1 功能描述

去噪模块在 Bayer 域进行，且四个通道强度分别可调，具有保留边缘和纹理的优点，并对随机噪声有一定的保留，提高整体图像信噪比。

8.7.2 API 参考

- [xmedia_isp_set_bnr_attr](#)：设置 bnr 属性参数。
- [xmedia_isp_get_bnr_attr](#)：获取 bnr 属性参数。

[xmedia_isp_set_bnr_attr](#)

【描述】

设置 bnr 参数。

【语法】

```
xmedia_s32 xmedia_isp_set_bnr_attr(xmedia_u32 pipe, const xmedia_isp_bnr_attr *bnr_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
bnr_attr	噪声抑制参数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

TNR 使能的最小分辨率为 64x64。

【举例】

无

【相关主题】

[xmedia_isp_get_bnr_attr](#)

xmedia_isp_get_bnr_attr

【描述】

获取噪声抑制参数。

【语法】

```
xmedia_s32 xmedia_isp_get_bnr_attr(xmedia_u32 pipe, xmedia_isp_bnr_attr *bnr_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
bnr_attr	噪声抑制参数。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【举例】

无

【相关主题】

[xmedia_isp_set_bnr_attr](#)

8.7.3 数据类型

- [xmedia_isp_bnr_center_mode](#): 定义 ISP NR 中心亮度的计算方式属性。
- [xmedia_isp_bnr_tnr_dp_mode](#): 定义 ISP NR 时域降噪的去坏点模式属性。
- [xmedia_isp_bnr_coring_pos](#): 定义 ISP NR 的 coring 位置属性。
- [xmedia_isp_bnr_ainr_pos](#): 定义 ISP NR 接入 AINR 位置属性。
- [xmedia_isp_bnr_manual_attr](#): 定义 ISP NR 手动属性。
- [xmedia_isp_bnr_auto_attr](#): 定义 ISP NR 自动属性。

- `xmedia_isp_bnr_wdr_attr`: 定义 ISP 在 WDR 模式下属性。
- `xmedia_isp_bnr_attr`: 定义 ISP NR 属性。

xmedia_isp_bnr_center_mode

【说明】

定义 ISP NR 中心亮度的计算方式属性。

【定义】

```
typedef enum {
    XMEDIA_ISP_BNR_CENTER_MODE_MIN_VAL = 0,
    XMEDIA_ISP_BNR_CENTER_MODE_SMALL_WIN,
    XMEDIA_ISP_BNR_CENTER_MODE_BIG_WIN,
    XMEDIA_ISP_BNR_CENTER_MODE_MAX_VAL,
    XMEDIA_ISP_BNR_CENTER_MODE_MAX
} xmedia_isp_bnr_center_mode;
```

【成员】

成员名称	描述
<code>XMEDIA_ISP_BNR_CENTER_MODE_MIN_VAL</code>	取小框和大框计算出的 luma 两个中较小值作为中心点 luma。
<code>XMEDIA_ISP_BNR_CENTER_MODE_SMALL_WIN</code>	取小框计算出的 luma 作为中心点 luma。
<code>XMEDIA_ISP_BNR_CENTER_MODE_BIG_WIN</code>	取大框计算出的 luma 作为中心点 luma。
<code>XMEDIA_ISP_BNR_CENTER_MODE_MAX_VAL</code>	取小框和大框计算出的 luma 两个中较大值作为中心点 luma。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_bnr_tnr_dp_mode

【说明】

定义 ISP NR 时域降噪的去坏点模式属性。

【定义】

```
typedef enum {  
    XMEDIA_ISP_BNR_TNR_DP_MODE_ZERO = 0,  
    XMEDIA_ISP_BNR_TNR_DP_MODE_ONE,  
    XMEDIA_ISP_BNR_TNR_DP_MODE_MAX  
} xmedia_isp_bnr_tnr_dp_mode;
```

【成员】

成员名称	描述
XMEDIA_ISP_BNR_TNR_DP_MODE_ZERO	模式 0，一般情况使用
XMEDIA_ISP_BNR_TNR_DP_MODE_ONE	模式 1，极低照度使用，强度更大

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_bnr_coring_pos

【说明】

定义 ISP NR 的 coring 位置属性。

【定义】

```
typedef enum {  
    XMEDIA_ISP_BNR_CORING_POS_PRE = 0,  
    XMEDIA_ISP_BNR_CORING_POS_POST,  
    XMEDIA_ISP_BNR_CORING_POS_MAX  
} xmedia_isp_bnr_coring_pos;
```

【成员】

成员名称	描述
XMEDIA_ISP_BNR_CORING_POS_PRE	回叠的原始未降噪的 raw。
XMEDIA_ISP_BNR_CORING_POS_POST	回叠的 Bayer2d 的输入:当 AINR 开启时, 回叠的 AI 的结果, 当 AINR 关闭时回叠的原始未降噪的 raw。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_bnr_ainr_pos

【说明】

定义 ISP NR 接入 AINR 位置属性。

【定义】

```
typedef enum {
    XMEDIA_ISP_BNR_AINR_POS_OFF = 0,
    XMEDIA_ISP_BNR_AINR_POS_PRE,
    XMEDIA_ISP_BNR_AINR_POS_POST,
    XMEDIA_ISP_BNR_AINR_POS_MAX
} xmedia_isp_bnr_ainr_pos;
```

【成员】

成员名称	描述
XMEDIA_ISP_BNR_AINR_POS_OFF	没有 AINR 输入。
XMEDIA_ISP_BNR_AINR_POS_PRE	AINR 的结果作为 bayer2d 的输入。
XMEDIA_ISP_BNR_AINR_POS_POST	AINR 的结果取代 bayer2d 的结果作为 bayer3d 的输入。

【注意事项】

接入点位置选择 POST 时，BTNR 必须打开，不允许关闭

【相关数据类型及接口】

无

xmedia_isp_bnr_manual_attr

【说明】

定义 isp_bnr 手动属性。

【定义】

```
typedef struct {
    xmedia_u16 snr_coarse_str[XMEDIA_ISP_BAYER_PATTERN_NUM];
    xmedia_u8  snr_coring_wgt;
    xmedia_u16 snr_luma_nr_str[XMEDIA_ISP_BNR_POINT_MAX_COUNT];
    xmedia_isp_bnr_center_mode snr_center_luma_mode;
    xmedia_isp_bnr_center_mode snr_center_tnr_luma_mode;
    xmedia_u16  snr_wdr_comp_scl;
    xmedia_u16  tnr_noise_min;
    xmedia_u16  tnr_bayer_str;
    xmedia_u16  tnr_md_thd_low;
    xmedia_u16  tnr_md_thd_high;
    xmedia_u16  tnr_wgt_min;
    xmedia_u16  tnr_min_ratio;
    xmedia_u16  tnr_mot_bld_thd_low;
    xmedia_u16  tnr_mot_bld_thd_high;
    xmedia_u16  tnr_edge_md_thd_low;
    xmedia_u16  tnr_edge_md_thd_high;
    xmedia_u16  tnr_flat_md_scl;
    xmedia_u16  tnr_edge_md_scl;
    xmedia_u8   tnr_md_mode;
    xmedia_u16  tnr_luma_md_thd_low;
    xmedia_u16  tnr_luma_md_thd_high;
    xmedia_u16  tnr_dark_md_scl;
    xmedia_u16  tnr_bright_md_scl;
    xmedia_u16  tnr_mot_scl[XMEDIA_ISP_BNR_COLOR_NUM_MAX_COUNT];
    xmedia_u16  tnr_var_scl;
    xmedia_u16  tnr_mot_noise_luma_thd_low;
    xmedia_u16  tnr_mot_noise_luma_thd_high;
    xmedia_u16  tnr_mot_noise_dark_scl;
}
```

```

xmedia_u16 tnr_mot_noise_bright_scl;
xmedia_u16 tnr_mot_dpc_str;
xmedia_u16 tnr_mot_convrg_thd[XMEDIA_ISP_BNR_THD_MAX_COUNT];
xmedia_u16 tnr_mot_noise_scl;
xmedia_u16 tnr_mot_noise_scl_high;
xmedia_bool tnr_dpc_en;
xmedia_isp_bnr_tnr_dp_mode tnr_dp_mode;
xmedia_u16 tnr_dp_thd_rb;
xmedia_u16 tnr_dp_thd_g;
xmedia_u16 tnr_dp_luma_thd_low;
xmedia_u16 tnr_dp_luma_thd_high;
xmedia_u16 tnr_dp_thd_low_dark_scl;
xmedia_u16 tnr_dp_thd_high_dark_scl;
xmedia_u16 tnr_luma_ratio[XMEDIA_ISP_BNR_POINT_MAX_COUNT];
xmedia_u16 tnr_wdr_comp_scl;
} xmedia_isp_bnr_manual_attr;

```



说明

```

#define XMEDIA_ISP_BAYER_PATTERN_NUM 4
#define XMEDIA_ISP_BNR_POINT_MAX_COUNT 61
#define XMEDIA_ISP_BNR_COLOR_NUM_MAX_COUNT 3
#define XMEDIA_ISP_BNR_THD_MAX_COUNT 4

```

【成员】

成员名称	描述	取值范围	默认值
snr_coarse_str	不同通道下的去噪基准强度，值越大，降噪越强。强度范围为 0~3 倍，一倍为 256，通常为 1~1.8 倍。	[0,768]	256
snr_coring_wgt	全局噪声回叠强度，会和 noise_reserve_lut 共同作用，作为乘法因子乘在 noise_reserve_lut 表上，值越大回叠噪声越多。	[0,128]	0
snr_luma_nr_str	根据亮度划分 60 个区域调节去噪强度表，其中暗区划分更密集(由第 0~32 个点控制)可进行更精细调试。强度范围为 0~2 倍，一倍为 128。	[0,256]	128
snr_center_luma_mode	Bayer2d 中心点 luma 的计算方式选择。	[0,3]	0
snr_center_tnr_luma_mod	Bayer3d 中心点 luma 的计算方式选择，	[0,3]	0

成员名称	描述	取值范围	默认值
e	XM7206V11A 不支持。		
snr_wdr_comp_scl	WDR 模式下短帧噪声回叠强度。值越小，短帧噪声回叠越少， XM7606V13 不支持。	[0,1023]	0
tnr_noise_min	(不建议调试)	[0,4095]	256
tnr_bayer_str	Bayer TNR 强度，值越大，时域降噪越强，噪声越小。	[64,1024]	128
tnr_md_thd_low	运动检测结果小于 tnr_md_thd_low，则认为是静止区域。如果运动检测结果大于 tnr_md_thd_high，则认为是运动区域。	[0,1023]	32
tnr_md_thd_high		[0,1023]	64
tnr_wgt_min	(不建议调试)	[0,16383]	512
tnr_min_ratio	(不建议调试) XM7206V11A 不支持。	[0,16383]	256
tnr_mot_bld_thd_low	如果边缘强度小于 mot_bld_thd_low，使用大窗口 MD 检测结果。如果边缘强度大于 mot_bld_thd_high，使用小窗口 MD 检测结果， 如果边缘强度大于 mot_bld_thd_low 并且小于 mot_bld_thd_high，则通过大窗口 MD 检测结果/小窗口 MD 检测结果两者之间线性插值得到 MD 检测结果。	[0,1023]	16
tnr_mot_bld_thd_high		[0,1023]	32
tnr_edge_md_thd_low	根据边缘强度自适应调整运动检测结果： 调整后运动检测结果 = 调整前运动检测结果 * md_scl/256	[0,1023]	16
tnr_edge_md_thd_high		[0,1023]	32
tnr_flat_md_scl		[0,1023]	256
tnr_edge_md_scl		[0,1023]	256

成员名称	描述	取值范围	默认值
	值得到 md_scl。		
tnr_md_mode	运动检测模式(建议使用模式 0, 不建议调试)	[0,3]	0
tnr_luma_md_thd_low	根据亮度自适应调整运动检测结果: 调整后运动检测结果 = 调整前运动检测结果 * md_scl/256 亮度小于 luma_md_thd_low 则为暗区, 调节因子 md_scl = dark_md_scl 亮度大于 luma_md_thd_high, 则为亮区, 调节因子 md_scl = bright_md_scl 如果边缘强度大于 luma_md_thd_low 并且小于 luma_md_thd_high, 则通过 dark_md_scl/bright_md_scl 两者之间线性插值得到 md_scl。	[0,4095]	256
tnr_luma_md_thd_high		[0,4095]	512
tnr_dark_md_scl		[0,1023]	256
tnr_bright_md_scl		[0,1023]	256
tnr_mot_scl	运动检测结果全局调整参数。值越大, 运动检测结果越大。XM7606V13 只用到 tnr_mot_scl[0]; XM7206V11A 可分 R/G/B 三通道调节。	[0,1023]	176
tnr_var_scl	值越大, 运动噪声越大, 运动收敛加快。	[0,1023]	32
tnr_mot_noise_luma_thd_low	根据亮度自适应调整运动区域的噪声回叠强度: 调整后噪声回叠强度 = 调整前噪声回叠强度 * mot_noise_scl / 256 亮度小于 mot_noise_luma_thd_low, 则为暗区, 调节因子 mot_noise_scl = mot_noise_dark_scl 亮度大于 mot_noise_luma_thd_high, 则为亮区, 调节因子 mot_noise_scl = mot_noise_bright_scl 如果亮度大于 mot_noise_luma_thd_low 并且小于 mot_noise_luma_thd_high, 则通过 mot_noise_dark_scl/mot_noise_bright_scl	[0,4095]	128
tnr_mot_noise_luma_thd_high		[0,4095]	256
tnr_mot_noise_dark_scl		[0,1023]	256
tnr_mot_noise_bright_scl		[0,1023]	256

成员名称	描述	取值范围	默认值
	两者之间线性插值得到 mot_noise_scl。		
tnr_mot_dpc_str	运动区域去坏点强度。值越小，运动区域去坏点强度越大。	[0,1023]	64
tnr_mot_convrg_thd	根据运动收敛程度自适应调整噪声回叠强度：	[0,4095]	1024
tnr_mot_noise_scl	调整后噪声回叠强度 = 调整前噪声回叠强度 * noise_scl / 256。	[0,1023]	256
tnr_mot_noise_scl_high (XM7606V13 不支持)	对于 XM7606V13, 如果运动收敛程度小于 mot_convrg_thd[0], 调节因子 noise_scl = mot_noise_scl; 如果运动收敛程度大于 mot_convrg_thd[1], 调节因子 noise_scl = 256; 如果运动收敛程度大于 mot_convrg_thd[0] 并且小于 mot_convrg_thd[1], 则通过 mot_noise_scl 和 256 两者之间线性插值得到 noise_scl。 本版本 tnr_mot_convrg_thd[0], tnr_mot_convrg_thd[1]有效。 对于 XM7206V11A, 如果运动收敛程度小于 mot_convrg_thd[0], 调节因子 noise_scl = mot_noise_scl; 如果运动收敛程度大于 mot_convrg_thd[1] 且小于 mot_convrg_thd[2], 调节因子 noise_scl = mot_noise_scl_high; 如果运动收敛程度大于 mot_convrg_thd[3], 调节因子 noise_scl = 256; 区间 (mot_convrg_thd[0],mot_convrg_thd[1]),	[0,1023]	256

成员名称	描述	取值范围	默认值
	(mot_convrg_thd[2],mot_convrg_thd[3])则通过线性插值得到 noise_scl。		
tnr_dpc_en	dpc 使能开关。	[0,1]	1
tnr_dp_mode	去坏点模式。模式 1 的去坏点强度比模式 0 更大。	[0,1]	0
tnr_dp_thd_rb	R/B 通道去坏点阈值。值越小，去坏点强度越大。	[0,1023]	384
tnr_dp_thd_g	G 通道去坏点阈值。值越小，去坏点强度越大。	[0,1023]	384
tnr_dp_luma_thd_low	根据亮度自适应调整去坏点强度：调整后 $dp_thd_rb = \text{调整前 } dp_thd_rb * dp_thd0_scl/256;$ $\text{调整后 } dp_thd_g = \text{调整前 } dp_thd_g * dp_thd1_scl/256$ 亮度小于 dp_luma_thd_low，则为暗区，调节因子 dp_thd0_scl = dp_thd_low_dark_scl, dp_thd1_scl = dp_thd_high_dark_scl 亮度大于 dp_luma_thd_high，则为亮区，调节因子 dp_thd0_scl = 256, dp_thd1_scl = 256 如果亮度大于 dp_luma_thd_low 并且小于 dp_luma_thd_high，则通过 dp_thd_low_dark_scl/256 两者之间线性插值得到 dp_thd0_scl。通过 dp_thd_high_dark_scl/256 两者之间线性插值得到 dp_thd1_scl $dp_thd_low_dark_scl/dp_thd_high_dark_scl$ 越小，暗区去坏点强度越大。	[0,4095]	128
tnr_dp_luma_thd_high		[0,4095]	256
tnr_dp_thd_low_dark_scl		[0,256]	256
tnr_dp_thd_high_dark_scl		[0,256]	256
tnr_luma_ratio	根据亮度划分 60 个区域的噪声回叠强度，其中暗区划分更密集(由第 0~32 个点控制)可进行更精细调试。	[0,256]	128

成员名称	描述	取值范围	默认值
tnr_wdr_comp_scl	WDR 模式下短帧噪声回叠强度。值越小，短帧噪声越小。 XM7206V11A 不支持。	[0,1023]	512

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_bnr_attr](#)

[xmedia_isp_bnr_auto_attr](#)

【说明】

定义 isp_bnr 自动属性。

【定义】

```
typedef struct {
    xmedia_u8    iso_num;
    xmedia_u32   iso_level[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16
    snr_coarse_str[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_BAYER_PATTERN_NUM];
    xmedia_u8    snr_coring_wgt[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16
    snr_luma_nr_str[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_BNR_POINT_MAX_COUNT];
    xmedia\_isp\_bnr\_center\_mode snr_center_luma_mode[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia\_isp\_bnr\_center\_mode snr_center_tnr_luma_mode[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16   snr_wdr_comp_scl[XMEDIA_ISP_ISO_MAX_COUNT];
    // Bayer3D param
    xmedia_u16   tnr_noise_min[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16   tnr_bayer_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16   tnr_md_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16   tnr_md_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16   tnr_wgt_min[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16   tnr_min_ratio[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16   tnr_mot_bld_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16   tnr_mot_bld_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16   tnr_edge_md_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16   tnr_edge_md_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16   tnr_flat_md_scl[XMEDIA_ISP_ISO_MAX_COUNT];
}
```

```

xmedia_u16 tnr_edge_md_scl[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8 tnr_md_mode[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 tnr_luma_md_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 tnr_luma_md_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 tnr_dark_md_scl[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 tnr_bright_md_scl[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 tnr_mot_scl[XMEDIA_ISP_ISO_MAX_COUNT]
[XMEDIA_ISP_BNR_COLOR_NUM_MAX_COUNT];
xmedia_u16 tnr_var_scl[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 tnr_mot_noise_luma_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 tnr_mot_noise_luma_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 tnr_mot_noise_dark_scl[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 tnr_mot_noise_bright_scl[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 tnr_mot_dpc_str[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 tnr_mot_convg_thd[XMEDIA_ISP_ISO_MAX_COUNT]
[XMEDIA_ISP_BNR_THD_MAX_COUNT];
xmedia_u16 tnr_mot_noise_scl[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 tnr_mot_noise_scl_high[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_bool tnr_dpc_en[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_isp_bnr_tnr_dp_mode tnr_dp_mode[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 tnr_dp_thd_rb[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 tnr_dp_thd_g[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 tnr_dp_luma_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 tnr_dp_luma_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 tnr_dp_thd_low_dark_scl[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 tnr_dp_thd_high_dark_scl[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16
tnr_luma_ratio[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_BNR_POINT_MAX_COUNT];
xmedia_u16 tnr_wdr_comp_scl[XMEDIA_ISP_ISO_MAX_COUNT];
} xmedia_isp_bnr_auto_attr;

```



说明

宏定义如下:

```

#define XMEDIA_ISP_ISO_MAX_COUNT 16
#define XMEDIA_ISP_BAYER_PATTERN_NUM 4
#define XMEDIA_ISP_BNR_POINT_MAX_COUNT 61
#define XMEDIA_ISP_BNR_COLOR_NUM_MAX_COUNT 3
#define XMEDIA_ISP_BNR_THD_MAX_COUNT 4

```

【成员】

成员名称	描述	取值范围	默认值
iso_num	ISO 数量。		
Iso_level	ISO 等级。		
snr_coarse_str	不同通道下的去噪基准强度，值越大，降噪越强。强度范围为 0~3 倍，一倍为 256，通常为 1~1.8 倍	[0,768]	256
snr_coring_wgt	全局噪声回叠强度，会和 noise_reserve_lut 共同作用，作为乘法因子乘在 noise_reserve_lut 表上，值越大回叠噪声越多	[0,128]	0
snr_luma_nr_str	根据亮度划分 60 个区域调节去噪强度表，其中暗区划分更密集(由第 0~32 个点控制)可进行更精细调试。强度范围为 0~2 倍，一倍为 128	[0,256]	128
snr_center_luma_mode	Bayer2d 中心点 luma 的计算方式选择	[0,3]	0
snr_center_tnr_luma_mode	Bayer3d 中心点 luma 的计算方式选择， XM7206V11A 不支持。	[0,3]	0
snr_wdr_comp_scl	WDR 模式下短帧噪声回叠强度。值越小，短帧噪声回叠越少， XM7606V13 不支持。	[0,1023]	0
tnr_noise_min	(不建议调试)	[0,4095]	256
tnr_bayer_str	Bayer TNR 强度，值越大，时域降噪越强，噪声越小	[64,1024]	128
tnr_md_thd_low	运动检测结果小于 tnr_md_thd_low，则认为是静止区域。如果运动检测结果大于 tnr_md_thd_high，则认为是运动区域	[0,1023]	32
tnr_md_thd_high		[0,1023]	64
tnr_wgt_min	(不建议调试)	[0,16383]	512
tnr_min_ratio	(不建议调试) XM7206V11A 不支持	[0,16383]	256
tnr_mot_bld_thd_low	如果边缘强度小于 mot_bld_thd_low，使用大窗口 MD 检测结果。如果边缘强度大于	[0,1023]	16
tnr_mot_bld_thd_high		[0,1023]	32

成员名称	描述	取值范围	默认值
	mot_bld_thd_high, 使用小窗口 MD 检测结果, 如果边缘强度大于 mot_bld_thd_low 并且小于 mot_bld_thd_high, 则通过大窗口 MD 检测结果/小窗口 MD 检测结果两者之间线性插值得到 MD 检测结果		
tnr_edge_md_thd_low	根据边缘强度自适应调整运动检测结果: 调整后运动检测结果 = 调整前运动检测结果 * md_scl/256 如果边缘强度小于 edge_md_thd_low, 则为平坦区域, 调节因子 md_scl = flat_md_scl 如果边缘强度大于 edge_md_thd_high, 则为边缘区域, 调节因子 md_scl = edge_md_scl 如果边缘强度大于 edge_md_thd_low 并且小于 edge_md_thd_high, 则通过 flat_md_scl/edge_md_scl 两者之间线性插值得到 md_scl	[0,1023]	16
tnr_edge_md_thd_high		[0,1023]	32
tnr_flat_md_scl		[0,1023]	256
tnr_edge_md_scl		[0,1023]	256
tnr_md_mode	运动检测模式(建议使用模式 0, 不建议调试)	[0,3]	0
tnr_luma_md_thd_low	根据亮度自适应调整运动检测结果: 调整后运动检测结果 = 调整前运动检测结果 * md_scl/256 亮度小于 luma_md_thd_low 则为暗区, 调节因子 md_scl = dark_md_scl 亮度大于 luma_md_thd_high, 则为亮区, 调节因子 md_scl = bright_md_scl 如果边缘强度大于 luma_md_thd_low 并且小于 luma_md_thd_high, 则通过 dark_md_scl/bright_md_scl 两者之间线性插值得到 md_scl	[0,4095]	256
tnr_luma_md_thd_high		[0,4095]	512
tnr_dark_md_scl		[0,1023]	256
tnr_bright_md_scl		[0,1023]	256
tnr_mot_scl	运动检测结果全局调整参数。值越大, 运动	[0,1023]	176

成员名称	描述	取值范围	默认值
	检测结果越大。 XM7606V13 只用到 tnr_mot_scl[0]; XM7206V11A 可分 R/G/B 三通道调节。		
tnr_var_scl	值越大, 运动噪声越大, 运动收敛加快。	[0,1023]	32
tnr_mot_noise_luma_thd_low	根据亮度自适应调整运动区域的噪声回叠强度: 调整后噪声回叠强度 = 调整前噪声回叠强度 * mot_noise_scl / 256	[0,4095]	128
tnr_mot_noise_luma_thd_high		[0,4095]	256
tnr_mot_noise_dark_scl		[0,1023]	256
tnr_mot_noise_bright_scl		[0,1023]	256
	亮度小于 mot_noise_luma_thd_low, 则为暗区, 调节因子 mot_noise_scl = mot_noise_dark_scl 亮度大于 mot_noise_luma_thd_high, 则为亮区, 调节因子 mot_noise_scl = mot_noise_bright_scl 如果亮度大于 mot_noise_luma_thd_low 并且小于 mot_noise_luma_thd_high, 则通过 mot_noise_dark_scl/mot_noise_bright_scl 两者之间线性插值得到 mot_noise_scl		
tnr_mot_dpc_str	运动区域去坏点强度。值越小, 运动区域去坏点强度越大。	[0,1023]	64
tnr_mot_convrg_thd (XM7606V13 只用到 tnr_mot_convrg_thd[0]和 tnr_mot_convrg_thd[1])	根据运动收敛程度自适应调整噪声回叠强度: 调整后噪声回叠强度 = 调整前噪声回叠强度 * noise_scl / 256。	[0,4095]	1024
tnr_mot_noise_scl		[0,1023]	256
tnr_mot_noise_scl_high (XM7606V13 不支持)		[0,1023]	256
	对于 XM7606V13 , 如果运动收敛程度小于 mot_convrg_thd[0], 调节因子 noise_scl = mot_noise_scl; 如果运动收敛程度大于 mot_convrg_thd[1], 调节因子 noise_scl = 256; 如果运动收敛程度大于 mot_convrg_thd[0]并且小于 mot_convrg_thd[1], 则通过 mot_noise_scl 和 256 两者之间线性插值得		

成员名称	描述	取值范围	默认值
	到 noise_scl。 本版本 tnr_mot_convrg_thd[0], tnr_mot_convrg_thd[1]有效。 对于 XM7206V11A, 如果运动收敛程度小于 mot_convrg_thd[0], 调节因子 noise_scl = mot_noise_scl; 如果运动收敛程度大于 mot_convrg_thd[1]且小于 mot_convrg_thd[2], 调节因子 noise_scl = mot_noise_scl_high; 如果运动收敛程度大于 mot_convrg_thd[3], 调节因子 noise_scl = 256; 区间(mot_convrg_thd[0],mot_convrg_thd[1]), (mot_convrg_thd[2],mot_convrg_thd[3])则通过线性插值得到 noise_scl。		
tnr_dpc_en	dpc 使能开关。	[0,1]	1
tnr_dp_mode	去坏点模式。模式 1 的去坏点强度比模式 0 更大。	[0,1]	0
tnr_dp_thd_rb	R/B 通道去坏点阈值。值越小, 去坏点强度越大	[0,1023]	384
tnr_dp_thd_g	G 通道去坏点阈值。值越小, 去坏点强度越大	[0,1023]	384
tnr_dp_luma_thd_low	根据亮度自适应调整去坏点强度: 调整后 $dp_thd_rb = \text{调整前 } dp_thd_rb * dp_thd0_scl/256$; 调整后 $dp_thd_g = \text{调整前 } dp_thd_g * dp_thd1_scl/256$ 亮度小于 dp_luma_thd_low, 则为暗区, 调节因子 $dp_thd0_scl = dp_thd_low_dark_scl$, $dp_thd1_scl = dp_thd_high_dark_scl$	[0,4095]	128
tnr_dp_luma_thd_high		[0,4095]	256
tnr_dp_thd_low_dark_scl		[0,256]	256
tnr_dp_thd_high_dark_scl		[0,256]	256

成员名称	描述	取值范围	默认值
	亮度大于 dp_luma_thd_high, 则为亮区, 调节因子 dp_thd0_scl = 256, dp_thd1_scl = 256 如果亮度大于 dp_luma_thd_low 并且小于 dp_luma_thd_high, 则通过 dp_thd_low_dark_scl/256 两者之间线性插值得到 dp_thd0_scl。通过 dp_thd_high_dark_scl/256 两者之间线性插值得到 dp_thd1_scl dp_thd_low_dark_scl/dp_thd_high_dark_scl 越小, 暗区去坏点强度越大		
tnr_luma_ratio	根据亮度划分 60 个区域的噪声回叠强度, 其中暗区划分更密集(由第 0~32 个点控制)可进行更精细调试.	[0,256]	128
tnr_wdr_comp_scl	WDR 模式下短帧噪声回叠强度。值越小, 短帧噪声越小。 XM7206V11A 不支持。	[0,1023]	512

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_bnr_attr](#)

[xmedia_isp_set_bnr_attr](#)

[xmedia_isp_get_bnr_attr](#)

[xmedia_isp_bnr_wdr_attr](#)

【说明】

定义 isp_bnr 在 wdr 模式下属性, 与自动/手动模式并行。

【定义】

```
typedef struct {
    xmedia_u8 wdr_frame_str[XMEDIA_ISP_WDR_CHN_MAX_NUM];
```

```
} xmedia_isp_bnr_wdr_attr;
```

说明

宏定义如下：

```
#define XMEDIA_ISP_WDR_CHN_MAX_NUM    2
```

【成员】

成员名称	描述
wdr_frame_str	WDR 模式下每一合成帧中长短帧分别对应的去噪强度。 wdr_frame_str[0]是长帧去噪强度，wdr_frame_str[1]是短帧去噪强度，值越大，去噪强度越大。

【注意事项】

注意各参数的取值范围。

【相关数据类型及接口】

[xmedia_isp_bnr_attr](#)

[xmedia_isp_set_bnr_attr](#)

[xmedia_isp_get_bnr_attr](#)

xmedia_isp_bnr_attr

【说明】

定义 isp_bnr 属性。

【定义】

```
typedef struct {  
    xmedia_bool            enable;  
    xmedia\_isp\_bnr\_ainr\_pos ainr_pos_switch;  
    xmedia\_isp\_bnr\_coring\_pos coring_pos_switch;  
    xmedia_bool            noise_reserve_enable;  
    xmedia_u16             noise_reserve_lut[XMEDIA_ISP_BNR_POINT_MAX_COUNT];  
    xmedia_u32             noise_profile_thd;  
    xmedia_double          noise_profile_coef0[XMEDIA_ISP_BNR_CALIB_POINT_MAX_COUNT];  
    xmedia_double          noise_profile_coef1[XMEDIA_ISP_BNR_CALIB_POINT_MAX_COUNT];  
    xmedia_bool            snr_wdr_comp_en;  
    xmedia_bool            tnr_enable;
```

```

xmedia_bool          tnr_luma_comp_enable;
xmedia_bool          tnr_wdr_comp_enable;
xmedia_video_operation_mode op_type;
xmedia_isp_bnr_auto_attr auto_attr;
xmedia_isp_bnr_manual_attr manual_attr;
xmedia_isp_bnr_wdr_attr wdr;
} xmedia_isp_bnr_attr;

```



说明

宏定义如下：

```
#define XMEDIA_ISP_BNR_POINT_MAX_COUNT 61
```

```
#define XMEDIA_ISP_BNR_CALIB_POINT_MAX_COUNT 5
```

【成员】

成员名称	描述
enable	BNR 使能开关。
ainr_pos_switch	AINR 结果的接入位置。
coring_pos_switch	噪声回叠的内容选择。
noise_reserve_enable	是否进行噪声保留。
noise_reserve_lut	根据亮度划分 60 个区域的噪声回叠强度。其中暗区划分更密集(由第 0~32 个点控制)可进行更精细调试
noise_profile_thd	标定系数的 iso 阈值。
noise_profile_coef0	ISO 小于 noise_profile_thd 的标定系数。
noise_profile_coef1	ISO 大于 noise_profile_thd 的标定系数。
snr_wdr_comp_en	WDR 模式下短帧噪声回叠强度控制功能使能。 XM7606V13 不支持。
tnr_enable	是否进行 bayer3d 去噪。 wdr 模式下，不允许开启 b3d。
tnr_luma_comp_enable	根据亮度分段曲线调整噪声回叠强度使能。
tnr_wdr_comp_enable	WDR 模式下短帧噪声回叠强度控制功能使能。 XM7206V11A 不支持。
op_type	操作模式。

成员名称	描述
auto_attr	自动模式。
manual_attr	手动模式。
wdr	BNR WDR 下模式参数。

【注意事项】

AINR 开关使能时，BNR 必须使能，不允许关闭。

ainr_pos_switch 设置为 POST 时，tnr_enable 必须开启，不允许关闭。

【相关数据类型及接口】

[xmedia_isp_set_bnr_attr](#)

[xmedia_isp_get_bnr_attr](#)

8.8 AINR

8.8.1 功能描述

AINR 模块用于控制 AI 降噪使能开关以及微调降噪效果。

8.8.2 API 参考

- [xmedia_isp_load_ainr_model](#): 加载 AINR 模型。
- [xmedia_isp_unload_ainr_model](#): 卸载 AINR 模型。
- [xmedia_isp_set_ainr_attr](#): 设置 AINR 属性。
- [xmedia_isp_get_ainr_attr](#): 获取 AINR 属性。
- [xmedia_isp_get_ainr_state](#): 获取 AINR 运行状态。
- [xmedia_isp_load_model](#): 加载 AI ISP 模型。
- [xmedia_isp_unload_model](#): 卸载 AI ISP 模型。
- [xmedia_isp_bind_model](#): 绑定 ISP 和 AI 模型。
- [xmedia_isp_unbind_model](#): 解绑 ISP 和 AI 模型。

xmedia_isp_load_ainr_model

【描述】

加载 AINR 模型。

【语法】

```
xmedia_s32 xmedia_isp_load_ainr_model(xmedia_u32 pipe, const xmedia_void
*model[XMEDIA_ISP_AINR_MODEL_MAX_NUM]);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
model	存储 AINR 模型的 buffer 的首地址。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

#define AINR_MODEL_MAX_NUM 8

#define XMEDIA_ISP_AINR_MODEL_MAX_NUM AINR_MODEL_MAX_NUM

XMEDIA_ISP_AINR_MODEL_MAX_NUM 为能加载最大模型数，跟芯片版本相关。

模型与 SENSOR 型号，分辨率相关。

模型加载操作，需先关闭 ainr 功能。

模型不允许重复加载。

【举例】

无

【相关主题】

[xmedia_isp_unload_ainr_model](#)

xmedia_isp_unload_ainr_model

【描述】

卸载 ainr 模型。

【语法】

```
xmedia_s32 xmedia_isp_unload_ainr_model(xmedia_u32 pipe);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

模型卸载操作，需先关闭 ainr 功能。

模型不允许重复卸载。

【举例】

无

【相关主题】

[xmedia_isp_load_ainr_model](#)

xmedia_isp_set_ainr_attr

【描述】

设置 ainr 属性

【语法】

```
xmedia_s32 xmedia_isp_set_ainr_attr(xmedia_u32 pipe, const xmedia\_isp\_ainr\_attr *ainr_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ainr_attr	ainr 属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

AINR 使能，需先加载模型。

AINR 使能时，不允许加载、卸载模型。

【举例】

无

【相关主题】

[xmedia_isp_get_ainr_attr](#)

xmedia_isp_get_ainr_attr

【描述】

获取 ainr 属性。

【语法】

```
xmedia_s32 xmedia_isp_get_ainr_attr(xmedia_u32 pipe, xmedia\_isp\_ainr\_attr *ainr_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ainr_attr	ainr 属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【举例】

无

【相关主题】

[xmedia_isp_set_ainr_attr](#)

xmedia_isp_get_ainr_state

【描述】

获取 AINR 运行状态。

【语法】

```
xmedia_s32 xmedia_isp_get_ainr_state(xmedia_u32 pipe, xmedia_bool *state);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
state	ainr 真实开关状态。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

XM7606V13 不支持

【举例】

无

【相关主题】

xmedia_isp_load_model

【描述】

加载 AI ISP 模型。

【语法】

```
xmedia_s32 xmedia_isp_load_model(const xmedia_void *model, xmedia_handle *handle);
```

【参数】

参数名称	描述	输入/输出
model	存储 AINR 模型的 buffer 的首地址。	输入
handle	AI 模型返回的句柄。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

不支持动态加载模型

XM7606V13 不支持

【举例】

无

【相关主题】

[xmedia_isp_unload_model](#)

xmedia_isp_unload_model

【描述】

卸载 AI ISP 模型。

【语法】

```
xmedia_s32 xmedia_isp_unload_model(xmedia_handle handle);
```

【参数】

参数名称	描述	输入/输出
handle	AI 模型的句柄。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

不支持动态卸载。

XM7606V13 不支持

【举例】

无

【相关主题】

[xmedia_isp_load_model](#)

xmedia_isp_bind_model

【描述】

绑定 ISP 和 AI 模型。

【语法】

```
xmedia_s32 xmedia_isp_bind_model(xmedia_u32 pipe,  
xmedia_handle handle[XMEDIA_ISP_AI_MODEL_MAX_NUM]);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
handle	模型对应的句柄。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

不支持动态绑定。

```
#define XMEDIA_ISP_AI_MODEL_MAX_NUM    (XMEDIA_ISP_ISO_MAX_COUNT  
* XMEDIA_ISP_AI_MODEL_PER_ISO)
```

XM7206V11A XMEDIA_ISP_AI_MODEL_PER_ISO 等于 1。

XM7606V13 不支持

【举例】

无

【相关主题】

[xmedia_isp_unbind_model](#)

xmedia_isp_unbind_model

【描述】

解绑 ISP 和 AI 模型。

【语法】

```
xmedia_s32 xmedia_isp_unbind_model(xmedia_u32 pipe,  
xmedia_handle handle[XMEDIA_ISP_AI_MODEL_MAX_NUM]);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
handle	AI 模型对应的句柄。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

不支持动态解绑

XM7606V13 不支持

【举例】

无

【相关主题】

[xmedia_isp_bind_model](#)

8.8.3 数据类型

- [xmedia_isp_ainr_manual_attr](#): 定义 AINR 手动模式属性。
- [xmedia_isp_ainr_auto_attr](#): 定义 AINR 自动模式属性。

- [xmedia_isp_ainr_attr](#): 定义 ISP AINR 属性。

xmedia_isp_ainr_manual_attr

【说明】

定义 AINR 手动模式属性。

【定义】

```
typedef struct {  
    xmedia_u16  pos_gain;  
    xmedia_u16  gau_gain;  
} xmedia_isp_ainr_manual_attr;
```

【成员】

成员名称	描述
pos_gain	AINR 控制降噪强度参数，影响整体降噪强度，值越高，降噪效果越强，默认值为 64，取值范围[0,1023]，一般情况不建议调整。
gau_gain	AINR 控制降噪强度参数，主要影响暗区降噪表现，值越高，降噪效果越强，默认值为 64，取值范围[0,1023]。

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_ainr_attr](#)

xmedia_isp_ainr_auto_attr

【说明】

定义 AINR 自动模式属性。

【定义】

```
typedef struct {  
    xmedia_u8    iso_num;  
    xmedia_u32   iso_level[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u16   pos_gain [XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u16   gau_gain [XMEDIA_ISP_ISO_MAX_COUNT];  
} xmedia_isp_ainr_auto_attr;
```

【成员】

成员名称	描述
iso_num	ISO 数目。
iso_level	ISO 等级。
pos_gain	AINR 控制降噪强度参数，影响整体降噪强度，值越高，降噪效果越强，默认值为 64，取值范围[0,1023]，一般情况不建议调整。
gau_gain	AINR 控制降噪强度参数，主要影响暗区降噪表现，值越高，降噪效果越强，默认值为 64，取值范围[0,1023]。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_ainr_attr

【说明】

定义 ISP AINR 属性。

【定义】

```
typedef struct {  
    xmedia_bool          enable;  
    xmedia_u32            switch_thd_low;  
    xmedia_u32            switch_thd_high;  
    xmedia_video_operation_mode op_type;  
    xmedia_isp_ainr_manual_attr manual_attr;  
    xmedia_isp_ainr_auto_attr auto_attr;  
} xmedia_isp_ainr_attr;
```

【成员】

成员名称	描述
enable	ainr 功能开关。
switch_thd_low	AINR 开关控制低阈值

成员名称	描述
switch_thd_high	AINR 开关控制高阈值
op_type	操作模式。 0: 自动 1: 手动
manual_attr	手动模式参数。
auto_attr	自动模式参数。

【注意事项】

- switch_thd_low 应低于 switch_thd_high。
- AINR 模块 enable 关闭时，AINR 固定关闭。
- AINR 模块 enable 使能时，底层 AINR 真实开关随当前系统 ISO 联动，当系统 ISO 低于 switch_thd_low 时，AINR 自动关闭；当系统 ISO 高于 switch_thd_high 时，AINR 自动开启；当系统 ISO 处于两者之间时，底层 AINR 开关维持上一帧状态不变。此功能用于避免 AINR 模块在开关临界 ISO 附近来回振荡导致频繁开启/关闭。

【相关数据类型及接口】

无

8.9 Dehaze

8.9.1 功能描述

dehaze 是通过动态的改变图象的对比度和亮度来实现的，将图像分块，统计每块内的像素值，估算出雾的浓度，根据局部自适应曲线调整去雾强度。dehaze 分为手动和自动模式，两种模式下均可调节去雾强度。

8.9.2 API 参考

- [xmedia_isp_set_dehaze_attr](#)：设置去雾属性。
- [xmedia_isp_get_dehaze_attr](#)：获取去雾属性。

xmedia_isp_set_dehaze_attr

【描述】

设置 dehaze 属性

【语法】

```
xmedia_s32 xmedia_isp_set_dehaze_attr(xmedia_u32 pipe, const xmedia_isp_dehaze_attr
*dehaze_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
dehaze_attr	去雾属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【举例】

无

【相关主题】

[xmedia_isp_get_dehaze_attr](#)

xmedia_isp_get_dehaze_attr

【描述】

获取 dehaze 属性。

【语法】

```
xmedia_s32 xmedia_isp_get_dehaze_attr(xmedia_u32 pipe, xmedia_isp_dehaze_attr
*dehaze_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
dehaze_attr	去雾属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【举例】

无

【相关主题】

[xmedia_isp_set_dehaze_attr](#)

8.9.3 数据类型

- [xmedia_isp_dehaze_manual_attr](#): 定义去雾手动模式属性。
- [xmedia_isp_dehaze_auto_attr](#): 定义去雾自动模式属性。
- [xmedia_isp_dehaze_attr](#): 定义 ISP 去雾属性。

xmedia_isp_dehaze_manual_attr

【说明】

定义去雾手动模式属性。

【定义】

```
typedef struct {
    xmedia_u8 strength;
    xmedia_u16 bmap_curve[XMEDIA_ISP_DEHAZE_MAP_LENGTH];
    xmedia_u16 dmap_curve[XMEDIA_ISP_DEHAZE_MAP_LENGTH];
} xmedia_isp_dehaze_manual_attr;
```



说明

宏定义如下：

```
#define XMEDIA_ISP_DEHAZE_MAP_LENGTH 17
```

【成员】

成员名称	描述
strength	透雾强度值，取值范围[0, 255]。 XM7606V13 不支持。
bmap_curve	控制暗处区域亮度补偿曲线，低照度场景，强度开弱以避免增强彩色噪声。取值范围[0,511]
dmap_curve	控制亮处区域色度补偿曲线，与 LCE 搭配，发生 contour 则降低强度。取值范围[0,511]

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_dehaze_attr](#)

xmedia_isp_dehaze_attr

【说明】

定义去雾自动模式属性。

【定义】

```
typedef struct {
    xmedia_u8 iso_num;
    xmedia_u32 iso_level[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8 strength[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16
bmap_curve[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_DEHAZE_MAP_LENGTH];
    xmedia_u16
dmap_curve[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_DEHAZE_MAP_LENGTH];
} xmedia_isp_dehaze_attr;
```

【成员】

成员名称	描述
iso_num	ISO 数目。
iso_level	ISO 等级。
strength	透雾强度值，取值范围[0, 255]。 XM7606V13 不支持。
bmap_curve	控制暗处区域亮度补偿曲线，低照度场景，强度开弱以避免增强彩色噪声。取值范围[0,511]
dmap_curve	控制亮处区域色度补偿曲线，与 LCE 搭配，发生 contour 则降低强度。取值范围[0,511]

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_dehaze_attr

【说明】

定义 ISP 去雾属性。

【定义】

```
typedef struct {
    xmedia_bool          enable;
    xmedia_bool          userlut_enable;
    xmedia_u8            dehaze_lut[XMEDIA_ISP_DEHAZE_LUT_NUM];
    xmedia_video_operation_mode op_type;
    xmedia_isp_dehaze_manual_attr manual_attr;
    xmedia_isp_dehaze_auto_attr auto_attr;
    xmedia_u16           tmprflt_incr_coef;
    xmedia_u16           tmprflt_decr_coef;
} xmedia_isp_dehaze_attr;
```



说明

宏定义如下:

```
#define XMEDIA_ISP_DEHAZE_LUT_NUM    256
```

【成员】

成员名称	描述
enable	使能 Dehaze 功能。
userlut_enable	用户自定义去雾曲线使能。 暂未支持
dehaze_lut	用户自定义去雾曲线。 暂未支持
op_type	操作模式。 0:自动 1: 手动
manual_attr	手动模式。
auto_attr	自动模式。
tmprflt_incr_coef	时域滤波递增系数, 值越大, 画面由暗到亮收敛速度越快, 取值范围[0,255]。
tmprflt_decr_coef	时域滤波递减系数, 值越大, 画面由亮到暗收敛速度越快, 取值范围[0,255]。

【注意事项】

无

【相关数据类型及接口】

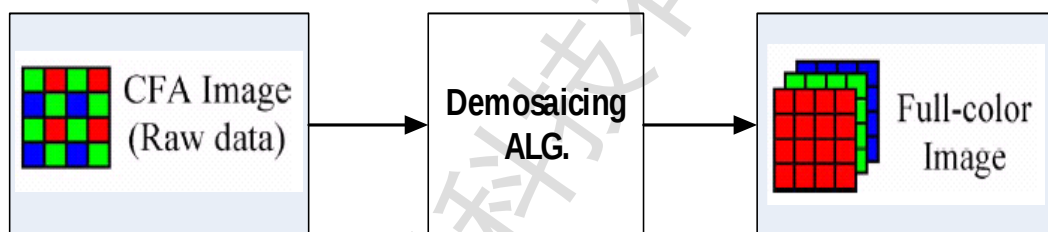
无

8.10 Demosaic

8.10.1 功能描述

demosaic 模块实现的功能是：将输入的 bayer 数据转化成 RGB 数据。CFA（Color Filter Array，彩色滤波阵列）得到的 bayer 数据，每一像素点只能获得 R、G、B 三基色中的一种彩色分量值。为获得彩色图像，需要利用当前像素及周围像素的色彩分量值，估计出当前点缺失的其他两个分量值（如图 8-3 所示）。

图8-3 去马赛克处理图



须知

在处理高频黑白图像时，由于去马赛克模块对输入 Bayer CFA 数据进行了插值，因此在高频区域容易发生混叠，导致出现打开去马赛克模块高频区域清晰度反而下降的情况，此为正常现象。

8.10.2 API 参考

- [xmedia_isp_set_demosaic_attr](#)：设置 demosaic 参数。
- [xmedia_isp_get_demosaic_attr](#)：获取 demosaic 参数。

xmedia_isp_set_demosaic_attr

【描述】

设置 Demosaic 参数。

【语法】

```
xmedia_s32 xmedia_isp_set_demosaic_attr(xmedia_u32 pipe, const xmedia\_isp\_demosaic\_attr *demosaic_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
demosaic_attr	Demosaic 参数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【举例】

无

【相关主题】

[xmedia_isp_get_demosaic_attr](#)

xmedia_isp_get_demosaic_attr

【描述】

获取 Demosaic 参数。

【语法】

```
xmedia_s32 xmedia_isp_get_demosaic_attr(xmedia_u32 pipe, xmedia\_isp\_demosaic\_attr
*demosaic_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
demosaic_attr	Demosaic 参数。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无。

【举例】

无。

【相关主题】

[xmedia_isp_set_demosaic_attr](#)

8.10.3 数据类型

- [xmedia_isp_demosaic_manual_attr](#): 定义 isp_demosaic 手动属性。
- [xmedia_isp_demosaic_auto_attr](#): 定义 isp_demosaic 自动属性。
- [xmedia_isp_demosaic_attr](#): 定义 isp_demosaic 属性。

xmedia_isp_demosaic_manual_attr

【说明】

定义 isp_demosaic 手动属性。

【定义】

```
typedef struct {  
    xmedia_u8  mf_de_str;  
    xmedia_u8  mf_de_osh_str;  
    xmedia_u8  mf_de_ush_str;  
    xmedia_u8  hf_de_str;  
    xmedia_u8  hf_smooth_range;  
    xmedia_u16 detail_smooth_str;  
    xmedia_u8  detail_mf_de_bias;  
    xmedia_u8  detail_hf_de_bias;  
} xmedia_isp_demosaic_manual_attr;
```

【成员】

成员名称	描述	取值范围	默认值
mf_de_str	中频增强强度，越强画面中频细节越清晰，但细节变粗，同时会加重黑白边线。	[0x0, 0xFF]	64
mf_de_osh_str	细节增强 overshoot 强度控制，越大画面白边越强。	[0x0, 0xFF]	255
mf_de_ush_str	细节增强 undershoot 强度控制，越大画面白边越强。	[0x0, 0xFF]	255
hf_de_str	高频增强强度，调高时画面高频细节增加，但相应噪声也会增强。	[0x0, 0xFF]	2
hf_smooth_range	细节平滑范围，数值越大，越多细节被滤至高频，hf_de_str 作用也越明显。	[0x1, 0x7]	1
detail_smooth_str	暂未支持。		
detail_mf_de_bias	细节区中频锐化强度偏置，细节区实际中频锐化强度为 mf_de_str-mf_de_bias。 XM7606V13 不支持。	[0x0, 0xFF]	0
detail_hf_de_bias	细节区高频锐化强度偏置，细	[0x0, 0xFF]	0

成员名称	描述	取值范围	默认值
	节区实际高频锐化强度为 hf_de_str-hf_de_bias。 XM7606V13 不支持。		

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_set_demosaic_attr](#)

[xmedia_isp_demosaic_auto_attr](#)

【说明】

定义 isp_demosaic 自动属性。

【定义】

```
typedef struct {  
    xmedia_u8  iso_num;  
    xmedia_u32 iso_level[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u8  mf_de_str[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u8  mf_de_osh_str[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u8  mf_de_ush_str[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u8  hf_de_str[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u8  hf_smooth_range[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u16 detail_smooth_str[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u8  detail_mf_de_bias[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u8  detail_hf_de_bias[XMEDIA_ISP_ISO_MAX_COUNT];  
} xmedia_isp_demosaic_auto_attr;
```

 说明

宏定义如下：

```
#define XMEDIA_ISP_ISO_MAX_COUNT    16
```

【成员】

成员名称	描述	取值范围	默认值
iso_num	ISO 等级。		

成员名称	描述	取值范围	默认值
iso_level	ISO 数量。		
mf_de_str	中频增强强度，越强画面中频细节越清晰，但细节变粗，同时会加重黑白边线。	[0x0, 0xFF]	64
mf_de_osh_str	细节增强 overshoot 强度控制，越大画面白边越强。	[0x0, 0xFF]	255
mf_de_ush_str	细节增强 undershoot 强度控制，越大画面白边越强。	[0x0, 0xFF]	255
hf_de_str	高频增强强度，调高时画面高频细节增加，但相应噪声也会增强。	[0x0, 0xFF]	2
hf_smooth_range	细节平滑范围，数值越大，越多细节被滤至高频，hf_de_str 作用也越明显。	[0x1, 0x7]	1
detail_smooth_str	暂未支持。		
detail_mf_de_bias	细节区中频锐化强度偏置，细节区实际中频锐化强度为 mf_de_str-mf_de_bias。 XM7606V13 不支持。	[0x0, 0xFF]	0
detail_hf_de_bias	细节区高频锐化强度偏置，细节区实际高频锐化强度为 hf_de_str-hf_de_bias。 XM7606V13 不支持。	[0x0, 0xFF]	0

【注意事项】

无

【相关数据类型及接口】

xmedia_isp_demosaic_attr

xmedia_isp_demosaic_attr

【说明】

定义 isp_demosaic 属性。

【定义】

```
typedef struct {
    xmedia_bool          enable;
    xmedia_video_operation_mode op_type;
    xmedia_isp_demosaic_auto_attr auto_attr;
    xmedia_isp_demosaic_manual_attr manual_attr;
} xmedia_isp_demosaic_attr;
```

【成员】

成员名称	描述	取值范围	默认值
enable	Demosaic 模块使能。	[0,1] 0: 禁止; 1: 使能;	0x1
op_type	Demosaic 工作类型。		
auto_attr	Demosaic 自动模式。		
manual_attr	Demosaic 手动模式。		

【注意事项】

DEMOSAIC 功能分为自动和手动参数：

- enable 为 XMEDIA_TRUE，op_type 为 XMEDIA_VIDEO_OPERATION_MODE_AUTO 时，使用自动 DEMOSAIC 参数。
- enable 设置为 XMEDIA_TRUE，op_type 为 XMEDIA_VIDEO_OPERATION_MODE_MANUAL 时，使用手动 DEMOSAIC 参数。

【相关数据类型及接口】

无。

8.11 黑电平

8.11.1 功能描述

黑电平通常指没有外界光线输入时，sensor 仍会输出的亮度值。ISP 需要减去这个亮度值，以进行颜色的处理。

8.11.2 API 参考

- [xmedia_isp_set_black_level_attr](#)：设置黑电平属性。
- [xmedia_isp_get_black_level_attr](#)：获取黑电平属性。

xmedia_isp_set_black_level_attr

【描述】

设置黑电平属性。

【语法】

```
xmedia_s32 xmedia_isp_set_black_level_attr(xmedia_u32 pipe, const xmedia_isp_blc_attr
*black_level);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
black_level	黑电平属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_isp.h

- 库文件: libxmedia_isp.a

【注意】

无

【举例】

无

【相关主题】

无

xmedia_isp_get_black_level_attr

【描述】

获取黑电平属性。

【语法】

```
xmedia_s32 xmedia_isp_get_black_level_attr(xmedia_u32 pipe, xmedia_isp_blc_attr* black_level);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
black_level	黑电平属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【举例】

无

【相关主题】

无

8.11.3 数据类型

- [xmedia_isp_blc_manual_attr](#): 定义 BLC 手动属性。
- [xmedia_isp_blc_auto_attr](#): 定义 BLC 自动属性。
- [xmedia_isp_blc_attr](#): 定义 BLC 属性。

xmedia_isp_blc_manual_attr

【说明】

定义 BLC 手动属性。

【定义】

```
typedef struct {  
    xmedia_u16 black_level[XMEDIA_ISP_BAYER_PATTERN_NUM];  
} xmedia_isp_blc_manual_attr;
```

【成员】

成员名称	描述
black_level	黑电平的值，分别表示 R、Gr、B、Gb 分量的黑电平值。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_blc_auto_attr

【说明】

定义 blc 自动属性。

【定义】

```
typedef struct {  
    xmedia_u8 iso_num;  
    xmedia_u32 iso_level[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u16 black_level  
[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_BAYER_PATTERN_NUM];  
} xmedia_isp_blc_auto_attr;
```

【成员】

成员名称	描述
iso_num	ISO 数量。
iso_level	ISO 等级。
black_level	黑电平的值，分别表示 R、Gr、B、Gb 分量的黑电平值。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_blc_attr

【说明】

定义 blc 属性。

【定义】

```
typedef struct {  
    xmedia_video_operation_mode op_type;  
    xmedia_isp_blc_manual_attr manual_attr;  
    xmedia_isp_blc_auto_attr auto_attr;  
} xmedia_isp_blc_attr;
```

【成员】

成员名称	描述
op_type	操作模式。
manual_attr	手动模式属性。
auto_attr	自动模式属性。

【注意事项】

无

【相关数据类型及接口】

无

8.12 去 FPN

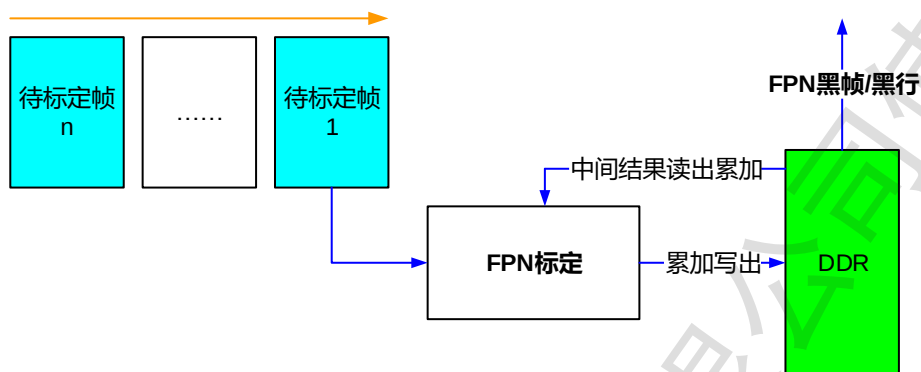
8.12.1 功能描述

Sensor 将光信号转换成电信号，再通过数百万个 ADC 器件后输出图像。每个像素结构中的光电二极管的尺寸、掺杂浓度、生产过程中的沾污以及 MOS 场效应管的参数的偏差等都会造成像素输出信号的变化，由于这些偏差造成的噪声对于给定的单个像素它是固定的，这种噪声就是固定模式噪声 FPN（Fixed Pattern Noise）。去 FPN 模块就是要把这些固定模式噪声消除。

FPN 的标定过程：

- 步骤 1 首先确定 FPN 在哪个 ISO 下最为严重，在此 ISO 下固定增益，关闭镜头的光圈，让 sensor 采集黑帧。
- 步骤 2 启动标定，当第一帧过来时，把满足标定条件的 RAW 数据写到内存，当第二帧过来时，通过读出端口把内存中的黑帧读出来与第二帧进行累加后再通过写出端口再写回内存，后面的帧也是这样，但是最后一帧写出时会对累加的黑帧平均后减去 BLC 偏置再写回内存。
- 步骤 3 当求平均写出后，标定结束。此时，内存中就把低照度下的固定列噪声存储下来了，然后，把该黑帧存储到外存上即可。

图8-4 FPN 标定示意图

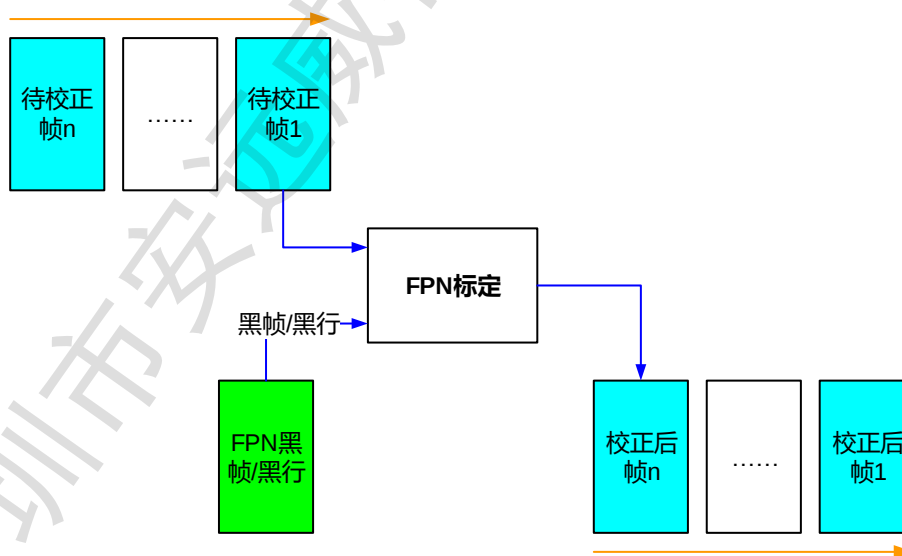


----结束

FPN 的校正过程：

- 步骤 1 把标定好的黑帧从外装载入到内存中。
- 步骤 2 当从 sensor 过采集过来的带 FPN 的正常图像过来后，用每帧都减去在内存中的黑帧，从而就得到了消除 FPN 后的校正帧。

图8-5 FPN 校正示意图



----结束

8.12.2 API 参考

- `xmedia_isp_set_fpn_calibrate`：设置去 FPN 标定属性。

- [xmedia_isp_set_fpn_attr](#): 设置去 FPN 属性。
- [xmedia_isp_get_fpn_attr](#): 获取去 FPN 属性。

xmedia_isp_set_fpn_calibrate

【描述】

设置去 FPN 标定属性。

【语法】

```
xmedia_s32 xmedia_isp_set_fpn_calibrate(xmedia_u32 pipe, const xmedia\_isp\_fpn\_calibrate\_attr
*calibrate_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
calibrate_attr	去 FPN 标定属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: `xmedia_isp.h`
- 库文件: `libxmedia_isp.a`

【注意】

非 PIPE1 通路，该标定功能，暂未支持。

【举例】

无

【相关主题】

无

xmedia_isp_set_fpn_attr

【描述】

设置去 FPN 属性。

【语法】

```
xmedia_s32 xmedia_isp_set_fpn_attr(xmedia_u32 pipe, const xmedia_isp_fpn_attr *fpn_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
fpn_attr	去 FPN 属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【举例】

无

【相关主题】

[xmedia_isp_get_fpn_attr](#)

xmedia_isp_get_fpn_attr

【描述】

获取去 FPN 属性。

【语法】

```
xmedia_s32 xmedia_isp_get_fpn_attr(xmedia_u32 pipe, xmedia_isp_fpn_attr *fpn_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
fpn_attr	去 FPN 属性指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

【举例】

无

【相关主题】

[xmedia_isp_set_fpn_attr](#)

8.12.3 数据类型

- [xmedia_isp_fpn_calibrate_attr](#): 定义去 FPN 的标定属性。
- [xmedia_isp_fpn_attr](#): 定义去 FPN 校正属性。

xmedia_isp_fpn_calibrate_attr

【说明】

定义去 FPN 的标定属性。

【定义】

```
typedef struct {  
    xmedia_u32      frame_num;  
    xmedia_u32      offset;  
    xmedia_video_frame_info fpn_frame;  
} xmedia_isp_fpn_calibrate_attr;
```

【成员】

成员名称	描述
frame_num	标定的帧数，需为 2 的整数次幂。 XM7606V13 取值范围：{4, 8, 16, 32, 64} XM7206V11A 取值范围：{4, 8, 16, 32, 64, 128, 256}
offset	黑帧所有像素的平均值。
fpn_frame	标定出来的黑帧信息。

【注意事项】

XM7606V13:

FPN 在线标定接口单次调用最大支持连续 64 帧标定，针对 64 帧以上的 FPN 标定，需要多次调用标定接口，具体使用方式可参见 fpn 标定 sample。

XM7206V11A:

FPN 在线标定接口单次调用最大支持连续 64 帧标定。

【相关数据类型及接口】

[xmedia_isp_set_fpn_calibrate](#)

[xmedia_isp_fpn_attr](#)

xmedia_isp_fpn_attr

【说明】

定义去 FPN 校正属性。

【定义】

```
typedef struct {
    xmedia_bool          enable;
    xmedia_video_frame_info  fpn_frm_info;
} xmedia_isp_fpn_attr;
```

【成员】

成员名称	描述
enable	FPN 校正使能。
fpn_frm_info	黑帧帧信息。

【注意事项】

无

【相关数据类型及接口】

无

8.13 GCAC

8.13.1 功能描述

色差(chromatic aberration)是指光学上透镜无法将各种波长的光聚焦在同一点上的现象，是一种与镜头有关的缺陷，它产生的主要原因是不同波长的光具有不同的折射率（色散现象）。色差可以分为如下的两类：

• 轴向色差(axial chromatic aberration)

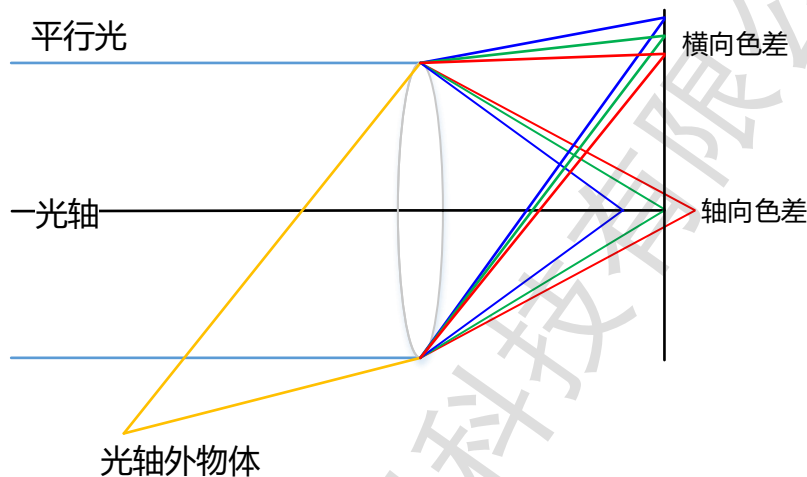
- 透镜对于不同的光会有不同的折射率，因此不同颜色的光会聚焦在不同的焦平面上，大口径镜头容易产生这种色差，缩小光圈可以减弱轴向色差。
- 人眼对于 G 通道更敏感，一般 G 通道可以正确对焦，从而引起 R、B 的模糊，造成高光区与低光区交界处出现明显的紫边表现。
- 具有明显的局部特性，因此校正紫边时采用 local CAC。

• 横向色差(lateral chromatic aberration)

- 透镜的放大倍数也与折射率有关，它使得不同波长光线的像高不同，即不同波长的光会聚焦在焦平面上不同的位置，会造成 R、G、B 3 通道具有不同的影像高度，在影像上产生色的错位。

- 越偏离图像中心，横向色差越明显，一般横向色差表现为物体相对的两侧边缘出现不同的颜色，但具体表现为什么颜色与镜头组密切相关，不同的镜头组会表现出不同种类的颜色边缘。
- 具有全局特性，在校正时采用 Global CAC，而不同镜头的校正参数由标定得到。

图8-6 色差图解



8.13.2 API 参考

- `xmedia_isp_set_gcac_attr`: 设置横向色差校正参数。
- `xmedia_isp_get_gcac_attr`: 获取横向色差校正参数。

`xmedia_isp_set_gcac_attr`

【描述】

设置横向色差校正参数。

【语法】

```
xmedia_s32 xmedia_isp_set_gcac_attr(xmedia_u32 pipe, const xmedia_isp_gcac_attr *gcac_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
gcac_attr	横向色差校正参数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_isp.h
- 库文件：libxmedia_isp.a

【注意】

无。

【举例】

无。

【相关主题】

[xmedia_isp_get_gcac_attr](#)

xmedia_isp_get_gcac_attr

【描述】

获取横向色差校正参数。

【语法】

```
xmedia_s32 xmedia_isp_get_gcac_attr(xmedia_u32 pipe, xmedia\_isp\_gcac\_attr *gcac_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
gcac_attr	横向色差校正参数。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无。

【相关主题】

[xmedia_isp_set_gcac_attr](#)

8.13.3 数据类型

- [xmedia_isp_gcac_attr](#): 横向色差校正属性。

xmedia_isp_gcac_attr

【说明】

定义横向色差校正属性。

【定义】

```
typedef struct {
    xmedia_bool enable;
    xmedia_u16  ver_coord;
    xmedia_u16  hor_coord;
    xmedia_s16  param_red_a;
    xmedia_s16  param_red_b;
    xmedia_s16  param_red_c;
    xmedia_s16  param_blue_a;
    xmedia_s16  param_blue_b;
    xmedia_s16  param_blue_c;
    xmedia_u8   ver_norm_shift;
    xmedia_u8   ver_norm_factor;
    xmedia_u8   hor_norm_shift;
    xmedia_u8   hor_norm_factor;
}
```

```

xmedia_u16 cor_var_thd;
} xmedia_isp_gcac_attr;

```

【成员】

成员名称	描述
enable	横向色差校正使能
ver_coord	光学中心纵坐标值。
hor_coord	光学中心横坐标。
param_red_a	R 通道对应的半径多项式的系数 a。
param_red_b	R 通道对应的半径多项式的系数 b。
param_red_c	R 通道对应的半径多项式的系数 c。
param_blue_a	B 通道对应的半径多项式的系数 a。
param_blue_b	B 通道对应的半径多项式的系数 b。
param_blue_c	B 通道对应的半径多项式的系数 c。
ver_norm_shift	垂直方向的归一化移位参数。
ver_norm_factor	垂直方向的归一化系数。
hor_norm_shift	水平方向的归一化移位参数。
hor_norm_factor	水平方向的归一化参数。
cor_var_thd	横向色差校正的方差阈值， 暂未支持 。

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_set_gcac_attr](#)

[xmedia_isp_get_gcac_attr](#)

8.14 LCAC

8.14.1 功能描述

Local CAC 主要用来消除图像中出现的紫边问题，**仅在 XM7206V11A 版本中生效。**

8.14.2 API 参考

- `xmedia_isp_set_lcac_attr`: 设置去紫边参数。
- `xmedia_isp_get_lcac_attr`: 获取去紫边参数。

`xmedia_isp_set_lcac_attr`

【描述】

设置去紫边参数。

【语法】

```
xmedia_s32 xmedia_isp_set_lcac_attr(xmedia_u32 pipe, const xmedia_isp_lcac_attr *lcac_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
lcac_attr	去紫边参数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

头文件: `xmedia_isp.h`

库文件: `libxmedia_isp.a`

【注意】

无。

【举例】

无。

【相关主题】

[xmedia_isp_get_lcac_attr](#)

xmedia_isp_get_lcac_attr

【描述】

获取去紫边参数。

【语法】

```
xmedia_s32 xmedia_isp_get_lcac_attr(xmedia_u32 pipe, xmedia\_isp\_lcac\_attr *lcac_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
lcac_attr	去紫边参数。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

头文件: xmedia_isp.h

库文件: libxmedia_isp.a

【注意】

无。

【相关主题】

[xmedia_isp_set_lcac_attr](#)

8.14.3 数据类型

- [xmedia_isp_lcac_attr](#): 定义 LCAC 属性。
- [xmedia_isp_lcac_auto_attr](#): 定义 LCAC 自动属性。
- [xmedia_isp_lcac_manual_attr](#): 定义 LCAC 手动属性。

[xmedia_isp_lcac_attr](#)

【说明】

定义 isp_lcac 属性。

【定义】

```
typedef struct {  
    xmedia_bool          enable;  
    xmedia_video_operation_mode op_type;  
    xmedia_isp_lcac_auto_attr auto_attr;  
    xmedia_isp_lcac_manual_attr manual_attr;  
} xmedia_isp_lcac_attr;
```

【成员】

成员名称	描述	取值范围	默认值
enable	LCAC 模块使能。	[0,1] 0: 禁止; 1: 使能;	0x1
op_type	LCAC 工作类型。		
auto_attr	LCAC 自动模式。		
manual_attr	LCAC 手动模式。		

【注意事项】

LCAC 功能分为自动和手动参数:

enable 为 XMEDIA_TRUE, op_type 为

XMEDIA_VIDEO_OPERATION_MODE_AUTO 时, 使用自动 LCAC 参数。

enable 设置为 XMEDIA_TRUE, op_type 为

XMEDIA_VIDEO_OPERATION_MODE_MANUAL 时, 使用手动 LCAC 参数。

【相关数据类型及接口】

无。

xmedia_isp_lcac_auto_attr

【说明】

定义 isp_lcac 自动属性。

【定义】

```
typedef struct {  
    xmedia_u8  iso_num;  
    xmedia_u32 iso_level[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_s16  purple_upper_limit[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_s16  purple_lower_limit[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u8  edge_thd[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u8  de_purple_cr_str[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u8  de_purple_cb_str[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u8  luma_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u8  luma_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u8  luma_str_high[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u8  luma_str_low[XMEDIA_ISP_ISO_MAX_COUNT];  
} xmedia_isp_lcac_auto_attr;
```



说明

宏定义如下：

```
#define XMEDIA_ISP_ISO_MAX_COUNT 16
```

【成员】

成员名称	描述	取值范围	默认值
iso_num	ISO 等级		
iso_level	ISO 数量		
purple_upper_limit	要去除紫边颜色上限，圈定色相上限。	[-1024, 1023]	512
purple_lower_limit	要去除紫边颜色下限，圈定色相下限。	[-1024, 1023]	-512
edge_thd	边缘判断阈值	[0x0, 0xFF]	30
de_purple_cr_str	CR 通道去紫边强度	[0x0, 0xFF]	64

成员名称	描述	取值范围	默认值
de_purple_cb_str	CB 通道去紫边强度	[0x0, 0xFF]	64
luma_thd_high	判定高亮区阈值	[0x0, 0xFF]	200
luma_thd_low	判定非高亮区阈值	[0x0, 0xFF]	100
luma_str_high	高亮区降紫边强度	[0x0, 0xFF]	128
luma_str_low	非高亮区降紫边强度	[0x0, 0xFF]	0

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_lcac_attr](#)

xmedia_isp_lcac_manual_attr

【说明】

定义 isp_lcac 手动属性。

【定义】

```
typedef struct {
    xmedia_s16  purple_upper_limit;
    xmedia_s16  purple_lower_limit;
    xmedia_u8   edge_thd;
    xmedia_u8   de_purple_cr_str;
    xmedia_u8   de_purple_cb_str;
    xmedia_u8   luma_thd_high;
    xmedia_u8   luma_thd_low;
    xmedia_u8   luma_str_high;
    xmedia_u8   luma_str_low;
} xmedia_isp_lcac_manual_attr;
```

【成员】

成员名称	描述	取值范围	默认值
purple_upper_limit	要去除紫边颜色上限， 圈	[-1024, 1023]	512

成员名称	描述	取值范围	默认值
	定色相上限。		
purple_lower_limit	要去除紫边颜色下限，圈定色相下限。	[-1024, 1023]	-512
edge_thd	边缘判断阈值。	[0x0, 0xFF]	30
de_purple_cr_str	CR 通道去紫边强度。	[0x0, 0xFF]	64
de_purple_cb_str	CB 通道去紫边强度。	[0x0, 0xFF]	64
luma_thd_high	判定高亮区阈值。	[0x0, 0xFF]	200
luma_thd_low	判定非高亮区阈值。	[0x0, 0xFF]	100
luma_str_high	高亮区降紫边强度。	[0x0, 0xFF]	128
luma_str_low	非高亮区降紫边强度。	[0x0, 0xFF]	0

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_lcac_attr](#)

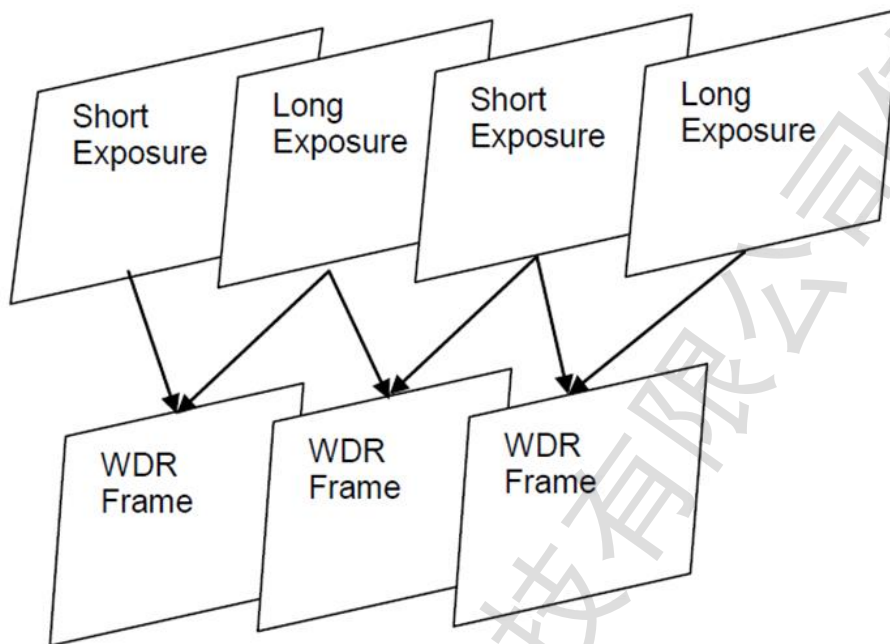
8.15 WDR

8.15.1 功能描述

由于传统的图像传感器的动态范围比较低，因此，如果用传统的图像传感器拍摄高动态范围的场景，就会要么亮处过曝，丢失很多细节；要么暗处曝光不足，细节难以分辨。为了记录高动态范围场景的每一个细节，可以采用多次曝光图像合成的方法。

WDR 算法的目的就是将从 sensor 输出的长短帧数据合成为一帧高 bit 的数据。如图 8-7 所示，长短帧数据是先缓存在 DDR 中，然后同时给 WDR 模块进行合成。合成的原则是择优选取，即对于比较亮的区域，采用短帧数据，而对于比较暗的区域，采用长帧数据。

图8-7 WDR 算法合成模型



•包括以下功能:

- 可以去除鬼影。通过 [xmedia_isp_wdr_mot_merge_mode](#) 来选择融合区域是采用短帧还是长帧;
- 可以通过 short_thd 和 long_thd 来控制选择长短帧的范围。即对于大于 short_thd 的像素选择短帧, 小于 long_thd 的像素选择长帧, 而对于在两者之间的像素采用长短帧融合的方式;
- 可以通过 md_thd_hig_gain, md_thd_slope_gain 来控制运动检测;
- 可以通过 short_expo_chk 来控制 WDR 融合是否检查短帧数据大小。

8.15.2 API 参考

- [xmedia_isp_set_wdr_attr](#): 设置 WDR 帧合成参数。
- [xmedia_isp_get_wdr_attr](#): 获取 WDR 帧合成参数。

xmedia_isp_set_wdr_attr

【描述】

设置帧合成参数。

【语法】

```
xmedia_s32 xmedia_isp_set_wdr_attr(xmedia_u32 pipe, const xmedia\_isp\_wdr\_attr *wdr_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
wdr_attr	帧合成参数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【举例】

无

【相关主题】

[xmedia_isp_get_wdr_attr](#)

xmedia_isp_get_wdr_attr

【描述】

获取帧合成参数。

【语法】

```
xmedia_s32 xmedia_isp_get_wdr_attr(xmedia_u32 pipe, xmedia\_isp\_wdr\_attr *wdr_attr)
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
wdr_attr	帧合成参数。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【举例】

无

【相关主题】

[xmedia_isp_set_wdr_attr](#)

8.15.3 数据类型

- [xmedia_isp_wdr_mot_merge_mode](#): 定义 ISP WDR 运动区融合属性。
- [xmedia_isp_wdr_manual_attr](#): 定义 ISP WDR 手动属性。
- [xmedia_isp_wdr_auto_attr](#): 定义 ISP WDR 自动属性。
- [xmedia_isp_wdr_mdt_attr](#): 定义 ISP WDR 运动检测属性。
- [xmedia_isp_wdr_combine_attr](#): 定义 ISP WDR 属性。
- [xmedia_isp_wdr_wbgain_position](#): 定义白平衡增益在 ISP 的位置属性。
- [xmedia_isp_wdr_attr](#): 定义 ISP WDR 属性。

xmedia_isp_wdr_mot_merge_mode

【说明】

定义 ISP WDR 运动区融合属性。

【定义】

```
typedef enum {  
    XMEDIA_ISP_MOT_MERGE_MODE_SHORT = 0,  
    XMEDIA_ISP_MOT_MERGE_MODE_LONG  = 1,  
    XMEDIA_ISP_MOT_MERGE_MODE_MAX  
} xmedia_isp_wdr_mot_merge_mode;
```

【成员】

成员名称	描述
XMEDIA_ISP_MOT_MERGE_MODE_SHORT	运动区融合短帧。
XMEDIA_ISP_MOT_MERGE_MODE_LONG	运动区融合长帧。

【相关数据类型及接口】

[xmedia_isp_wdr_mdt_attr](#)

xmedia_isp_wdr_manual_attr

【说明】

定义 WDR 手动模式属性。

【定义】

```
typedef struct {  
    xmedia_u8   md_thd_slope_gain;  
    xmedia_u16  md_thd_hig_gain;  
} xmedia_isp_wdr_manual_attr;
```

【成员】

成员名称	描述	取值范围
md_thd_slope_gain	手动模式下判断是否运动的斜率系数。值越大，运动程度越小。	[3,63]
md_thd_hig_gain	手动模式下判断是否运动的高阈值系数。值越	[256,4095]

成员名称	描述	取值范围
	大，运动程度越小。	

【相关数据类型及接口】

[xmedia_isp_wdr_mdt_attr](#)

xmedia_isp_wdr_auto_attr

【说明】

定义 WDR 自动属性。

【定义】

```
typedef struct {
    xmedia_u8    iso_num;
    xmedia_u32   iso_level[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8    md_thd_slope_gain[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16   md_thd_hig_gain[XMEDIA_ISP_ISO_MAX_COUNT];
} xmedia_isp_wdr_auto_attr;
```

【成员】

成员名称	描述	取值范围
iso_num	ISO 数目。	[1,16]
iso_level	ISO 等级。	[100,3276800]
md_thr_slope_gain	自动模式下判断是否运动的斜率系数。值越大，运动程度越小。	[3,63]
md_thr_hig_gain	自动模式下判断是否运动的高阈值系数。值越大，运动程度越小。	[256,4095]

【相关数据类型及接口】

[xmedia_isp_wdr_mdt_attr](#)

xmedia_isp_wdr_mdt_attr

【说明】

定义 ISP WDR 运动检测属性。

【定义】

```
typedef struct {  
    xmedia_isp_wdr_mot_merge_mode mot_merge_mode;  
    xmedia_bool mot_merge_short_ref_en;  
    xmedia_bool short_expo_chk;  
    xmedia_u16 short_check_thd;  
    xmedia_bool md_ref_flicker;  
    xmedia_u8 md_slope;  
    xmedia_u8 md_full_thd;  
    xmedia_u8 md_long_blend;  
    xmedia_video_operation_mode op_type;  
    xmedia_isp_wdr_manual_attr manual_attr;  
    xmedia_isp_wdr_auto_attr auto_attr;  
    xmedia_bool short_to_long_en;  
    xmedia_u16 short_to_long_short_thd;  
    xmedia_u16 short_to_long_short_slope;  
    xmedia_u16 short_to_long_long_thd;  
    xmedia_u8 short_to_long_rb_shift_mode;  
    xmedia_s16 short_to_long_red_shift;  
    xmedia_s16 short_to_long_blue_shift;  
    xmedia_u16 short_to_long_wb_gain_r;  
    xmedia_u16 short_to_long_wb_gain_b;  
    xmedia_u16 short_to_long_wb_gain_thd;  
} xmedia_isp_wdr_mdt_attr;
```

【成员】

成员名称	描述	取值范围	默认值
mot_merge_mode	运动区融合模式。 XM7606V13 不支持	[0,1]	0
mot_merge_short_ref_en	运动区融合长帧时，参考短帧亮度使能，开启时稍缓解紫边。 暂未支持	[0,1]	1
short_expo_chk	短帧亮度检查使能。	[0,1]	0
short_check_thd	短帧强制切换长帧亮度阈值。	[0,4095]	100

md_ref_flicker	暂未支持	[0,1]	0
md_slope	从绝对运动区到静止区的过渡斜率，越大运动区越少。	[0,255]	255
md_full_thd	运动检测中判断为全运动的阈值，当运动程度大于该阈值时，判断该像素点为全运动。	[0,255]	220
md_long_blend	运动区域叠加长帧的权重，当该值为 0xFE 时，运动区域全部选择长帧。	[0,255]	0
op_type	WDR 工作类型。 OP_TYPE_AUTO：自动； OP_TYPE_MANUAL：手动。	[0,1]	0
manual_attr	WDR 手动参数，具体参见 xmedia_isp_wdr_manual_attr		
auto_attr	WDR 自动参数，具体参见 xmedia_isp_wdr_manual_attr		
short_to_long_en	短帧区域强制切换长帧使能。	[0,1]	1
short_to_long_short_thd	强制切换低阈值。	[0, 4095]	50
short_to_long_short_slope	强制切换斜率。	[0, 4095]	1024
short_to_long_long_thd	强制切换高阈值。	[0, 4095]	2400
short_to_long_rb_shift_mode	暂未支持	[0,2]	0
short_to_long_red_shift	过曝区域 R 通道矫正值，不建议调整，通常设置为 0。暂未支持	[-4095,4095]	0
short_to_long_blue_shift	过曝区域 B 通道矫正值，不建议调整，通常设置为 0。暂未支持	[-4095,4095]	0
short_to_long_wb_gain_r	暂未支持	[0, 4095]	4095

short_to_long_wb_gain_b	暂未支持	[0, 4095]	4095
short_to_long_wb_gain_thd	暂未支持	[0, 4095]	2048

【注意事项】

mot_merge_mode、mot_merge_short_ref_en、md_ref_flicker、short_to_long_rb_shift_mode、short_to_long_red_shift、short_to_long_blue_shift、short_to_long_wb_gain_r、short_to_long_wb_gain_b、short_to_long_wb_gain_thd 参数暂未支持。

【相关数据类型及接口】

[xmedia_isp_wdr_combine_attr](#)

xmedia_isp_wdr_combine_attr

【说明】

定义 ISP WDR 属性。

【定义】

```
typedef struct {
    xmedia_bool    dgn_ref_bypass;
    xmedia_bool    motion_comp;
    xmedia_u16     short_thd;
    xmedia_u16     long_thd;
    xmedia_u8      force_long;
    xmedia_bool    ce_force_long_en;
    xmedia_u16     ce_force_long_slope;
    xmedia_u16     ce_force_long_hig_thd;
    xmedia\_isp\_wdr\_mdt\_attr wdr_mdt;
} xmedia_isp_wdr_combine_attr;
```

【成员】

成员名称	描述	取值范围	默认值
dgn_ref_bypass	减黑电平衰减补偿,最大值拉到4095。暂未支持 0: 补偿	[0,1]	0

	1: 不补偿		
motion_comp	WDR 运动补偿使能。 0: 关闭; 1: 使能。 在某些室内日光灯场景下, 运动补偿会加重频闪带来的横条纹现象, 此时可以关闭运动补偿。	[0,1]	1
short_thd	短曝光门限值, 超过门限值的图像数据只选择短曝光数据。	[0,4095]	2800
long_thd	长曝光门限值, 低于门限值的图像数据只选择长曝光数据。	[0,4095]	2400
force_long	强制输出长帧使能。 暂未支持	[0,2]	0
ce_force_long_en	基于亮度矫正运动检测结果使能。	[0,1]	0
ce_force_long_slope	运动强度衰减速度, 越大, 被检测为运动的区域越少。	[1,511]	6
ce_force_long_hig_thd	亮度值小于该阈值的区域运动检测权重进行衰减。	[0,4095]	694
wdr_mdt	WDR 运动检测参数, 具体参见 xmedia_isp_wdr_mdt_attr。		

【注意事项】

dgn_ref_bypass、forcelong 参数暂未支持

【相关数据类型及接口】

[xmedia_isp_wdr_attr](#)

xmedia_isp_wdr_wbgain_position

【说明】

定义白平衡增益在 ISP 的位置属性。

【定义】

```
typedef enum {  
    XMEDIA_ISP_WDR_WBGAIN_POSITION_IN_DG1 = 0,  
    XMEDIA_ISP_WDR_WBGAIN_POSITION_IN_WB = 1,  
    XMEDIA_ISP_WDR_WBGAIN_POSITION_MAX  
} xmedia_isp_wdr_wbgain_position;
```

【成员】

成员名称	描述
XMEDIA_ISP_WDR_WBGAIN_POSITION_IN_DG1	白平衡增益配置在 WDR 合成前的 DG1 处。
XMEDIA_ISP_WDR_WBGAIN_POSITION_IN_WB	白平衡增益配置在 WB 处。

【相关数据类型及接口】

[xmedia_isp_wdr_attr](#)

xmedia_isp_wdr_attr

【说明】

定义 ISP WDR 属性。

【定义】

```
typedef struct {  
    xmedia_bool enable;  
    xmedia\_isp\_wdr\_wbgain\_position wdr_wbgain_position;  
    xmedia\_isp\_wdr\_combine\_attr wdr_combine;  
} xmedia_isp_wdr_attr;
```

【成员】

成员名称	描述	取值范围	默认值
enable	WDR 使能。	[0,1]	0

成员名称	描述	取值范围	默认值
wdr_wbgain_position	白平衡增益在 ISP 的位置。 0: 白平衡增益配置在 WDR 合成前的 DG1 处。 1: 白平衡增益配置在 WB 处。	[0,1]	0
wdr_combine	传统的 WDR 合成结构体		

【注意事项】

无

8.16 LCE

8.16.1 功能描述

LCE 是局域对比度增强的简称。该算法采用局域直方图均衡来增强图像局部对比度，提升暗区细节和亮区细节。

8.16.2 API 参考

- [xmedia_isp_set_lce_attr](#): 设置 lce 属性。
- [xmedia_isp_get_lce_attr](#): 获取 lce 属性。
- [xmedia_isp_get_lce_info](#): 获取 lce blending 曲线。

xmedia_isp_set_lce_attr

【描述】

设置 lce 属性。

【语法】

```
xmedia_s32 xmedia_isp_set_lce_attr(xmedia_u32 pipe, const xmedia\_isp\_lce\_attr *lce_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号	输入
lce_attr	局部对比度增强属性	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【举例】

无

【相关主题】

[xmedia_isp_get_lce_attr](#)

xmedia_isp_get_lce_attr

【描述】

获取 lce 属性。

【语法】

```
xmedia_s32 xmedia_isp_get_lce_attr(xmedia_u32 pipe, xmedia\_isp\_lce\_attr *lce_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
lce_attr	局部对比度增强属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【举例】

无

【相关主题】

[xmedia_isp_set_lce_attr](#)

xmedia_isp_get_lce_info

【描述】

获取 lce blending 曲线，用于显示。

【语法】

```
xmedia_s32 xmedia_isp_get_lce_info(xmedia_u32 pipe, xmedia\_isp\_lce\_info *lce_info);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
Ice_info	原图和 Ice 图的 blending 曲线生成，用于显示。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_isp.h
- 库文件：libxmedia_isp.a

【注意】

无

【相关主题】

无

8.16.3 数据类型

- [xmedia_isp_Ice_manual_attr](#)：定义 Ice 手动属性。
- [xmedia_isp_Ice_auto_attr](#)：定义 Ice 自动属性。
- [xmedia_isp_Ice_attr](#)：定义 Ice 属性。
- [xmedia_isp_Ice_info](#)：定义 Ice 信息。

xmedia_isp_Ice_manual_attr

【说明】

定义 Ice 手动属性。

【定义】

```
typedef struct {
```

```

xmedia_u8 contrast_str;
xmedia_s8 contrast_blc_slope[XMEDIA_ISP_LCE_CONTRAST_SLOPE_NUM];
xmedia_s8 contrast_wht_slope[XMEDIA_ISP_LCE_CONTRAST_SLOPE_NUM];
xmedia_u8 blending_fmax;
xmedia_u16 blending_sigma_div;
xmedia_u8 blending_mean;
xmedia_u8 blending_blc_str;
xmedia_u8 blending_blc_value;
xmedia_u8 blending_wht_str;
xmedia_u8 blending_wht_value;
xmedia_u16 black_luma_expand[XMEDIA_ISP_LCE_CONTRAST_BWLE_NUM];
xmedia_u16 white_luma_expand[XMEDIA_ISP_LCE_CONTRAST_BWLE_NUM];
xmedia_u16 flat_denoise_thd;
xmedia_u16 flat_denoise_str;
xmedia_u8 flat_filter_str[XMEDIA_ISP_LCE_FLAT_FILTER_SIZE];
xmedia_u8 halo_str;
xmedia_u8 tpr_incr_coef;
xmedia_u8 tpr_decr_coef;
} xmedia_isp_lce_manual_attr;

```



说明

宏定义如下：

```

#define XMEDIA_ISP_LCE_CONTRAST_SLOPE_NUM    2
#define XMEDIA_ISP_LCE_FLAT_FILTER_SIZE      9
#define XMEDIA_ISP_LCE_CONTRAST_BWLE_NUM    2

```

【成员】

成员名称	描述
contrast_str	对比度强度，参数越大，对比度强度越强。取值范围[0,63]
contrast_blc_slope	0:lce 图的直方图暗部的权重斜率控制，负数抬亮暗部亮度；正数压暗暗部亮度；邻域内暗像素越多 LOW_SLOPE[0]作用越明显； 1:当出现只有几个灰阶，且灰阶占比很大的场景时，如果需要很强对比度，可适当调节这个参数为正值，不满足灰阶情况不调值，保持和 LOW_SLOPE[0]一致； 取值范围[-127,127]

contrast_wht_slope	0:Ice 图的直方图亮部的权重斜率控制， 负数抬亮亮部亮度(暗部中亮部也抬亮)；正数压暗亮部亮度(暗部中亮部也压暗)；邻域内亮像素越多 HIGH_SLOPE[0]作用越明显； 1：当出现只有几个灰阶，且灰阶占比很大的场景时，如果需要很强对比度，可适当调节这个参数为正值，不满足灰阶情况不调值，保持和 HIGH_SLOPE[0]一致； 取值范围[-127,127]
blending_fmax	原图和 Ice 图融合系数 fmax，fmax 越大原图占比越大。 取值范围[0,255]
blending_sigma_div	原图和 Ice 高斯融合曲线方差，计算公式为 $SIGMA_DIVISION = 65535 / (2 * \sigma * \sigma)$，sigma 取值为(1~127)。
blending_mean	原图和 Ice 高斯融合峰值的亮度值，该值表示此亮度的原图占比最大，Ice 图占比最小。 取值范围[2,32767]
blending_blc_str	融合曲线暗部权重抑制，增大 strength,可抑制 Ice 图暗处过于亮拉起噪声，值越大抑制越强，原图暗处占比越大。 取值范围[0,63]
blending_blc_value	暗部权重抑制亮度范围值。 取值范围[0,255]
blending_wht_str	融合曲线亮部过曝抑制参数，增大 whtctrl_strength,可抑制 Ice 图亮处过于亮导致过曝，值越大抑制越强，原图亮处占比越大。 取值范围[0,63]
blending_wht_value	亮部噪声抑制亮度范围值。 取值范围[0,255]
black_luma_expand	极暗区微调，0-1023 为亮度，0-511 为强度，256 为基数倍数，小于 256 为压暗暗部，大于 256 为抬亮暗部。

	取值范围： [0,1023] [0,511]
white_luma_expand	极亮区微调，0-1023 为亮度，0-511 为强度，256 为基数倍数，小于 256 为压暗亮部，大于 256 为抬亮亮部。 取值范围： [0,1023] [0,511]
flat_denoise_thd	平坦区对比度强度抑制的阈值，计算方式为 $th = (tile_width * tile_height) / flat_denoise_thd$
flat_denoise_str	平坦区对比度强度抑制的强度，计算方式为 $str = contrast_str - flat_denoise_str$
flat_filter_str	平坦区 curve filter，缓解 contour 0 0 0 0 15 0 0 0 0 为完全不参考邻域块，1 1 1 1 8 1 1 1 1 为参考邻域块的一部分。 取值范围[0,15]
halo_str	光晕减弱： 0:close; 255:光晕抑制最强； 取值范围[0,255]
tpr_incr_coef	LCE 时域平滑参数； 画面由暗到亮变化时域滤波系数，值越大，当前帧像素比重越大。 取值范围[0,255]
tpr_decr_coef	LCE 时域平滑参数； 画面由亮到暗变化时域滤波系数，值越大，当前帧像素比重越大

	取值范围[0,255]
--	-------------

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_lce_auto_attr

【说明】

定义 lce 自动属性。

【定义】

```
typedef struct {
    xmedia_u8    iso_num;
    xmedia_u32 iso_level[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8    contrast_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_s8
contrast_blc_slope[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_LCE_CONTRAST_SLOPE_NUM];
    xmedia_s8
contrast_wht_slope[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_LCE_CONTRAST_SLOPE_NUM];
    xmedia_u8    blending_fmax[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16 blending_sigma_div[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8    blending_mean[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8    blending_blc_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8    blending_blc_value[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8    blending_wht_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8    blending_wht_value[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16
black_luma_expand[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_LCE_CONTRAST_BWLE_NUM];
    xmedia_u16
white_luma_expand[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_LCE_CONTRAST_BWLE_NUM];
    xmedia_u16 flat_denoise_thd[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16 flat_denoise_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8
```

```

flat_filter_str[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_LCE_FLAT_FILTER_SIZE];
xmedia_u8 halo_str[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8 tpr_incr_coef[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8 tpr_decr_coef[XMEDIA_ISP_ISO_MAX_COUNT];
} xmedia_isp_lce_auto_attr;

```



说明

宏定义如下：

```
#define XMEDIA_ISP_ISO_MAX_COUNT 16
```

【成员】

成员名称	描述
iso_num	ISO 数量。 取值范围[1,16]
iso_level	ISO 等级。 取值范围[100,3276800]
contrast_str	对比度强度，参数越大，对比度强度越强。 取值范围[0,63]
contrast_blc_slope	0:lce 图的直方图暗部的权重斜率控制，负数抬亮暗部亮度；正数压暗暗部亮度；邻域内暗像素越多 LOW_SLOPE[0]作用越明显； 1:当出现只有几个灰阶，且灰阶占比很大的场景时，如果需要很强对比度，可适当调节这个参数为正值，不满足灰阶情况不调值，保持和 LOW_SLOPE[0]一致； 取值范围[-127,127]
contrast_wht_slope	0:lce 图的直方图亮部的权重斜率控制，负数抬亮亮部亮度(暗部中亮部也抬亮)；正数压暗亮部亮度(暗部中亮部也压暗)；邻域内亮像素越多 HIGH_SLOPE[0]作用越明显； 1: 当出现只有几个灰阶，且灰阶占比很大的场景时，如果需要很强对比度，可适当调节这个参数为正值，不满足灰阶情况不调值，保持和 HIGH_SLOPE[0]一致； 取值范围[-127,127]

blending_fmax	原图和 lce 图融合系数 fmax, fmax 越大原图占比越大。 取值范围[0,255]
blending_sigma_div	原图和 lce 高斯融合曲线方差, 计算公式为 $SIGMA_DIVISION = 65535 / (2 * \sigma * \sigma)$, sigma 取值为(1~127)。 取值范围[2,32767]
blending_mean	原图和 lce 高斯融合峰值的亮度值, 该值表示此亮度的原图占比最大, lce 图占比最小。 取值范围[0,255]
blending_blc_str	融合曲线暗部权重抑制, 增大 strength,可抑制 lce 图暗处过于亮拉起噪声, 值越大抑制越强, 原图暗处占比越大。 取值范围[0,63]
blending_blc_value	暗部权重抑制亮度范围值。 取值范围[0,255]
blending_wht_str	融合曲线亮部过曝抑制参数, 增大 whtctrl_strength,可抑制 lce 图亮处过于亮导致过曝, 值越大抑制越强, 原图亮处占比越大。 取值范围[0,63]
blending_wht_value	亮部噪声抑制亮度范围值。 取值范围[0,255]
black_luma_expand	极暗区微调, 0-1023 为亮度, 0-511 为强度, 256 为基数倍数, 小于 256 为压暗暗部, 大于 256 为抬亮暗部。 取值范围 [0,1023] [0,511]

white_luma_expand	极亮区微调, 0-1023 为亮度, 0-511 为强度, 256 为基数 倍数, 小于 256 为压暗亮部, 大于 256 为抬亮亮部。 取值范围 [0,1023] [0,511]
flat_denoise_thd	平坦区对比度强度抑制的阈值, 计算方式为 $th=(tile_width*tile_height)/flat_denoise_thd$
flat_denoise_str	平坦区对比度强度抑制的强度, 计算方式为 $str=contrast_str-flat_denoise_str$
flat_filter_str	平坦区 curve filter, 缓解 contour 0 0 0 0 15 0 0 0 0 为完全不参考邻域块, 1 1 1 1 8 1 1 1 1 为参考邻域块的一部分。 取值范围[0,15]
halo_str	光晕减弱: 0:close; 255:光晕抑制最强; 取值范围[0,255]
tpr_incr_coef	LCE 时域平滑参数; 画面由暗到亮变化时域滤波系数, 值越大, 当前帧像素 比重越大。 取值范围[0,255]
tpr_decr_coef	LCE 时域平滑参数; 画面由亮到暗变化时域滤波系数, 值越大, 当前帧像素 比重越大。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_lce_attr

【说明】

定义 lce 属性。

【定义】

```
typedef struct {
    xmedia_bool          enable;
    xmedia_video_operation_mode op_type;
    xmedia_bool          input_di_mode_en;
    xmedia_bool          output_di_mode_en;
    xmedia_isp_lce_manual_attr manual_attr;
    xmedia_isp_lce_auto_attr auto_attr;
} xmedia_isp_lce_attr;
```



说明

宏定义如下：

```
#define XMEDIA_ISP_LCE_MAX_CURVE_POINT_NUM    33
```

【成员】

成员名称	描述
enable	lce 使能开关。
op_type	操作类型。 0：自动 1：手动
input_di_mode_en	输入数据为 yuv 时，dither 使能。
output_di_mode_en	输出数据 dither 使能。
manual_attr	手动属性。
auto_attr	自动属性。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_lce_info

【说明】

定义 lce 信息。

【定义】

```
typedef struct {
    xmedia_u16 blending_weight_curve[XMEDIA_ISP_LCE_MAX_CURVE_POINT_NUM];
} xmedia_isp_lce_info;
```

【成员】

成员名称	描述
blending_weight_curve	原图和 lce 图融合 curve，用于显示取值范围[0,256]

【注意事项】

无

【相关数据类型及接口】

无

8.17 CA

8.17.1 功能描述

CA 支持 YUV 域 颜色饱和度调整，这个模块下有两个模式，一个是 CA 模式，另外一个 CP 模式（热成像上色），工作的时候，两者只能二选一。暂未支持 CP 模式。

在 CA 模式下，通过下面的公式可以将一个像素点 (Y, U, V) 映射到另一个像素点 (Y', U', V') 。

$$Y'=Y;$$

$$U'=aU;$$

$$V'=aV;$$

其中 a 是转换系数，采用这组公式可以在一定程度上保持亮度和色调的恒定，对像素点的饱和度做一个调整。转换系数 a 和像素点亮度 Y 联系，就可以根据亮度的变化来

调整饱和度，达到局部调整饱和度的目的，亮处的颜色更鲜艳，暗处的色噪不明显。同时，转换系数 a 和 ISP 的 ISO 值联系，达到降低低照度下的暗处色噪的目的。

在 CP 模式下，热成像的图像只有亮度信息，该模式下通过亮度信息 Y 查找上色的色板，查找对应的 YUV 的值作为输出的值。其中，色板是通过 YUV 格式存储的。

8.17.2 API 参考

- [xmedia_isp_set_ca_attr](#): 设定 chroma adjust 属性。
- [xmedia_isp_get_ca_attr](#): 获取 chroma adjust 属性。

xmedia_isp_set_ca_attr

【描述】

设定 chroma adjust 属性。

【语法】

```
xmedia_s32 xmedia_isp_set_ca_attr(xmedia_u32 pipe, const xmedia\_isp\_ca\_attr *ca_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ca_attr	chroma adjust 属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无。

【举例】

无。

【相关主题】

[xmedia_isp_get_ca_attr](#)

xmedia_isp_get_ca_attr

【描述】

获取 chroma adjust 属性。

【语法】

```
xmedia_s32 xmedia_isp_get_ca_attr(xmedia_u32 pipe, xmedia\_isp\_ca\_attr *ca_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ca_attr	chroma adjust 属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无。

【举例】

无。

【相关主题】

[xmedia_isp_set_ca_attr](#)

8.17.3 数据类型

- [xmedia_isp_ca_yuv_manual_attr](#): 定义 ISP chroma adjust 手动属性。
- [xmedia_isp_ca_yuv_auto_attr](#): 定义 ISP chroma adjust 自动属性。
- [xmedia_isp_ca_attr](#): 定义 ISP chroma adjust 属性。

xmedia_isp_ca_yuv_manual_attr

【说明】

定义 ISP chroma adjust 手动属性。

【定义】

```
typedef struct {
    xmedia_u8 sat_str;
    xmedia_u16 sat_by_y[XMEDIA_ISP_CA_YUV_SAT_NUM];
    xmedia_u16 sat_by_c[XMEDIA_ISP_CA_YUV_SAT_NUM];
} xmedia_isp_ca_yuv_manual_attr;
```

【成员】

成员名称	描述	取值范围	默认值
sat_str	yuv 域饱和度调整全局强度	0-15	2
sat_by_y	根据亮度调节饱和度，64 为 1 倍，	[0,255]	64
sat_by_c	根据饱和程度调节饱和度，64 为 1 倍， 暂未支持	[0,255]	64

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_set_ca_attr](#)

[xmedia_isp_get_ca_attr](#)

xmedia_isp_ca_yuv_auto_attr

【说明】

定义 ISP chroma adjust 自动属性。

【定义】

```
typedef struct {  
    xmedia_u8  iso_num;  
    xmedia_u32 iso_level[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u8  sat_str[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u16  
sat_by_y[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_CA_YUV_SAT_NUM];  
    xmedia_u16  
sat_by_c[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_CA_YUV_SAT_NUM];  
} xmedia_isp_ca_yuv_auto_attr;
```

【成员】

成员名称	描述	取值范围	默认值
iso_num	ISO 数量	[1, 16]	
iso_level	ISO 等级	[100, 3276800]	
sat_str	yuv 域饱和度自动调整全局强度	0-15	2
sat_by_y	根据亮度调节饱和度，64 为 1 倍，	[0,255]	64
sat_by_c	根据饱和程度调节饱和度，64 为 1 倍， 暂未支持	[0,255]	64

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_set_ca_attr](#)

[xmedia_isp_get_ca_attr](#)

xmedia_isp_ca_attr

【说明】

定义 ISP chroma adjust 属性。

【定义】

```
typedef struct {  
    xmedia_bool          enable;  
    xmedia_video_operation_mode op_type;  
    xmedia_isp_ca_yuv_manual_attr manual_attr;  
    xmedia_isp_ca_yuv_auto_attr auto_attr;  
} xmedia_isp_ca_attr;
```

【成员】

成员名称	描述
enable	chroma adjust 使能 0: 关闭 1: 使能
op_type	0: 自动 1: 手动
manual_attr	手动饱和度全局强度
auto_attr	自动全局饱和度强度

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_set_ca_attr](#)

8.18 Radial Crop

8.18.1 功能描述

WDR 模式下, 对于鱼镜头, DRC 会将四周暗处拉亮, 最后呈现出的图像的整体效果不是很好, 因此新引入一个算法, 将设定半径之外的地方直接拉黑掉。由于半径之外的数据为 0, 带黑电平 FE 统计信息可以以此为根据对半径以外的点进行舍弃。

8.18.2 API 参考

- [xmedia_isp_set_rc_attr](#): 设定 Radial Crop 属性。
- [xmedia_isp_get_rc_attr](#): 获取 Radial Crop 属性。

xmedia_isp_set_rc_attr

【描述】

设定 Radial Crop 属性。

【语法】

```
xmedia_s32 xmedia_isp_set_rc_attr(xmedia_u32 pipe, const xmedia_isp_rc_attr *rc_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
rc_attr	Radial Crop 属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败, 其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

XM7206V11A 不支持。

【举例】

无。

【相关主题】

[xmedia_isp_get_rc_attr](#)

xmedia_isp_get_rc_attr

【描述】

获取 Radial Crop 属性。

【语法】

```
xmedia_s32 xmedia_isp_get_rc_attr(xmedia_u32 pipe, const xmedia_isp_rc_attr *rc_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
rc_attr	Radial Crop 属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败, 其值为错误码。

【需求】

- 头文件: xmedia_isp.h

- 库文件: libxmedia_isp.a

【注意】

XM7206V11A 不支持。

【举例】

无。

【相关主题】

[xmedia_isp_set_rc_attr](#)

8.18.3 数据类型

- [xmedia_isp_rc_point](#): radial crop 中心点坐标。
- [xmedia_isp_rc_attr](#): 定义 ISP Radial Crop 属性。

xmedia_isp_rc_point

【说明】

radial crop 中心点坐标。

【定义】

```
typedef struct {
    xmedia_s32 x;
    xmedia_s32 y;
} xmedia_isp_rc_point;
```

【成员】

成员名称	描述	取值范围	默认值
x	中心点坐标配置 x	[0, 8191]	当前通路行宽的一半。
y	中心点坐标配置 y	[0, 8191]	当前通路行高的一半。

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_set_rc_attr](#)

[xmedia_isp_get_rc_attr](#)

xmedia_isp_rc_attr

【说明】

定义 ISP Radial Crop 属性。

【定义】

```
typedef struct {  
    xmedia_bool        enable;  
    xmedia_isp_rc_point center_coor;  
    xmedia_u32          radius;  
} xmedia_isp_rc_attr;
```

【成员】

成员名称	描述	取值范围	默认值
enable	Radial Crop 使能	[0, 1]	0, 默认关闭。
center_coor	中心点的坐标配置	X : [0, 8191] Y : [0, 8191]	X: 当前通路行宽的一半; Y: 当前通路行高的一半。
radius	拉黑的半径, 到中心点的距离大于半径的区域直接拉黑	[0, 11586]	当前通路行高的一半。

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_set_rc_attr](#)

[xmedia_isp_get_rc_attr](#)

8.19 CSC

8.19.1 功能描述

色彩空间转换模块(color space conversion)，用来将 RGB 信号转换成符合标准的 YUV 信号。

8.19.2 API 参考

- [xmedia_isp_set_csc_attr](#)：设定 CSC（色彩空间转换）模块属性。
- [xmedia_isp_get_csc_attr](#)：获取 CSC（色彩空间转换）模块属性。

xmedia_isp_set_csc_attr

【描述】

设定 CSC（色彩空间转换）属性。

【语法】

```
xmedia_s32 xmedia_isp_set_csc_attr(xmedia_u32 pipe, const xmedia\_isc\_attr *csc_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
csc_attr	CSC 属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_isc.h
- 库文件：libxmedia_isc.a

【注意】

无

【举例】

无

【相关主题】

[xmedia_isp_get_csc_attr](#)

xmedia_isp_get_csc_attr

【描述】

获取 CSC（色彩空间转换）属性。

【语法】

```
xmedia_s32 xmedia_isp_get_csc_attr(xmedia_u32 pipe, xmedia_isp_csc_attr *csc_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
csc_attr	CSC 属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_isp.h
- 库文件：libxmedia_isp.a

【注意】

无。

【举例】

无。

【相关主题】

[xmedia_isp_set_csc_attr](#)

8.19.3 数据类型

- [xmedia_isp_csc_color_gamut](#): 定义通道色域属性。
- [xmedia_isp_csc_matrix](#): 定义色彩空间转换矩阵系数。
- [xmedia_isp_csc_attr](#): 定义 ISP 色彩空间转换模块属性。

[xmedia_isp_csc_color_gamut](#)

【说明】

定义通道色域属性。

【定义】

```
typedef enum {
    XMEDIA_ISP_CSC_COLOR_GAMUT_BT601 = 0,
    XMEDIA_ISP_CSC_COLOR_GAMUT_BT709,
    XMEDIA_ISP_CSC_COLOR_GAMUT_BT2020,
    XMEDIA_ISP_CSC_COLOR_GAMUT_USER,
    XMEDIA_ISP_CSC_COLOR_GAMUT_MAX
} xmedia_isp_csc_color_gamut;
```

【成员】

成员名称	描述
XMEDIA_ISP_CSC_COLOR_GAMUT_BT601	色域范围为 BT.601。
XMEDIA_ISP_CSC_COLOR_GAMUT_BT709	色域范围为 BT.709。
XMEDIA_ISP_CSC_COLOR_GAMUT_BT2020	色域范围为 BT.2020。 XM7606V13、XM7206V11A 暂未支持；
XMEDIA_ISP_CSC_COLOR_GAMUT_USER	用户自定义色域范围。

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_set_csc_attr](#)

[xmedia_isp_get_csc_attr](#)

xmedia_isp_csc_matrix

【说明】

定义色彩空间转换矩阵系数。

【定义】

```
typedef struct {  
    xmedia_s16 csc_idc[XMEDIA_ISP_CSC_INPUT_DC_NUM];  
    xmedia_s16 csc_odc[XMEDIA_ISP_CSC_OUTPUT_DC_NUM];  
    xmedia_s16 csc_coef[XMEDIA_ISP_CSC_MATRIX_SIZE];  
} xmedia_isp_csc_matrix;
```



说明

宏定义:

```
#define XMEDIA_ISP_CSC_INPUT_DC_NUM    3  
#define XMEDIA_ISP_CSC_OUTPUT_DC_NUM  3  
#define XMEDIA_ISP_CSC_MATRIX_SIZE    9
```

【成员】

成员名称	描述
csc_idc	取值范围: [-1024,1023] CSC 转换矩阵, 输入直流分量。 XM7606V13 暂未支持;
csc_odc	取值范围: [-1024,1023] CSC 转换矩阵, 输出直流分量。
csc_coef	取值范围: [-4096,4095] CSC 转换矩阵, 3x3 系数。

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_csc_attr](#)

xmedia_isp_csc_attr

【说明】

定义 ISP 色彩空间转换模块属性。

【定义】

```
typedef struct {  
    xmedia_bool          enable;  
    xmedia\_isp\_csc\_color\_gamut color_gamut;  
    xmedia_u8            hue;  
    xmedia_u8            luma;  
    xmedia_u8            contrast;  
    xmedia_u8            saturation;  
    xmedia_bool          limited_range_en;  
    xmedia_bool          ext_csc_en;  
    xmedia_bool          ct_mode_en;  
    xmedia\_isp\_csc\_matrix csc_matrix;  
} xmedia_isp_csc_attr;
```

【成员】

成员名称	描述	取值范围	默认值
enable	使能 CSC 功能。	[0,1] 0：禁止； 1：使能；	0
color_gamut	色彩空间类型枚举。 XM7606V13、XM7206V11A 暂未支持 BT2020	[0,3] 默认值 XMEDIA_ISP_CSC_COLOR_GAMUT_BT709	1
hue	目标 YUV 空间色调。	[0,100]	50
luma	目标 YUV 空间亮度。	[0,100]	50
contrast	目标 YUV 空间对比度。	[0,100]	50

成员名称	描述	取值范围	默认值
saturation	目标 YUV 空间饱和度。	[0,100]	50
limited_range_en	目标 YUV 空间动态范围选择。	[0,1] 0: 输出为 Full Range; 1: 输出为 Limited Range。	0
ext_csc_en	选择亮度调整模式。	[0,1] 1: 表示 luma 可以调节更宽的亮度变化;	1
ct_mode_en	选择输出的对比度调整模式。 XM7606V13、XM7206V11A 暂未支持;	[0,1] 1: 改变 contrast 画面亮度不会变化;	1
csc_matrix	色彩空间转换矩阵系数。		

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_set_csc_attr](#)

[xmedia_isp_get_csc_attr](#)

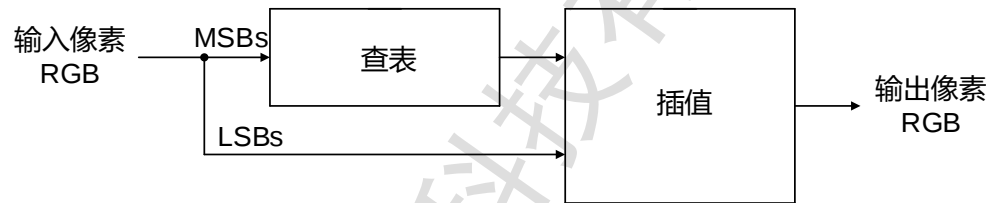
8.20 CLUT

8.20.1 功能描述

COLOR_3D_LUT 是利用线性 RGB 空间 17x17x17 分格的 3D LUT 实现复杂的颜色调整操作，比如亮度的调整，饱和度的调整，色调的调整。阴影区域，中间亮度区域，高亮区域分别调整。

它将像素的 RGB 信号的 bit 高位用来查找像素点位于表中的位置，利用 RGB 信号的 bit 低位用来在表的一个方格中插值出精确的偏移量。将偏移量应用于原像素值，得到调整后的最终像素值，从而使颜色得到调节。

图8-8 COLOR_3D_LUT 示意图



该单元通过查表的方式，获取 R、G 以及 B 分别的调节量 ΔR 、 ΔG 和 ΔB 。

总共需要三类查找表，R、G 和 B 各一张表。

CLUT 调节量处理的数据是[0, 4095]范围的 12bit 线性 RGB 数据。

8.20.2 API 参考

- [xmedia_isp_set_clut_attr](#)：设置 CLUT 属性。
- [xmedia_isp_get_clut_attr](#)：获取 CLUT 属性。

xmedia_isp_set_clut_attr

【描述】

设置 CLUT 属性。

【语法】

```
xmedia_s32 xmedia_isp_set_clut_attr(xmedia_u32 pipe, const xmedia_isp_clut_attr *clut_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
clut_attr	clut_attr 属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

XM7206V11A 不支持。

【举例】

无。

【相关主题】

[xmedia_isp_get_clut_attr](#)

xmedia_isp_get_clut_attr

【描述】

获取 CLUT 属性。

【语法】

```
xmedia_s32 xmedia_isp_get_clut_attr(xmedia_u32 pipe, xmedia\_isp\_clut\_attr *clut_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
clut_attr	clut_attr 属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

XM7206V11A 不支持。

【举例】

无。

【相关主题】

[xmedia_isp_set_clut_attr](#)

8.20.3 数据类型

- [xmedia_isp_clut_attr](#): CLUT 颜色色度和饱和度调整属性配置。
- [xmedia_isp_clut_manual_attr](#): clut 表手动属性。
- [xmedia_isp_clut_auto_attr](#): clut 表自动属性。

xmedia_isp_clut_attr

【说明】

定义 CLUT 的属性。

【定义】

```
typedef struct {  
    xmedia_bool          enable;  
    xmedia_video_operation_mode op_type;  
    xmedia_esp_clut_manual_attr manual_attr;  
    xmedia_esp_clut_auto_attr auto_attr;  
} xmedia_esp_clut_attr;
```

【成员】

成员名称	描述
enable	CLUT 使能。 取值范围：[0,1];
op_type	操作模式。
manual_attr	手动 clut 表。
auto_attr	自动 clut 表。

【注意事项】

XM7206V11A 不支持。

【相关数据类型及接口】

无

xmedia_esp_clut_manual_attr

【说明】

定义 CLUT 的手动属性。

【定义】

```
typedef struct {  
    xmedia_u16 lut[XMEDIA_ESP_CLUT_LUT_LENGTH];  
} xmedia_esp_clut_manual_attr;
```

【成员】

成员名称	描述
lut	手动 clut 表。取值范围：[0,1023]

【注意事项】

XM7206V11A 不支持。

【相关数据类型及接口】

无

xmedia_isp_clut_auto_attr

【说明】

定义 CLUT 的自动属性。

【定义】

```
typedef struct {
    xmedia_u8  ct_num;
    xmedia_u32 ct_level[XMEDIA_ISP_CLUT_CT_MAX_COUNT];
    xmedia_u32
lut[XMEDIA_ISP_CLUT_CT_MAX_COUNT][XMEDIA_ISP_CLUT_LUT_LENGTH];
} xmedia_isp_clut_auto_attr;
```

 说明

宏定义如下：

```
#define XMEDIA_ISP_CLUT_CT_MAX_COUNT      3
#define XMEDIA_ISP_CLUT_LUT_LENGTH      (4913*3)
```

【成员】

成员名称	描述
ct_num	色温数目。 取值范围：[1,3]
ct_level	色温等级。 取值范围： [0,65535]
lut	色彩调整参数，需通过额外的色彩调整工具进行参数 转换与建立。每一组 lut 与 ct_level 色温值——对应。 取值范围：[0,1023]

【注意事项】

XM7206V11A 不支持。

【相关数据类型及接口】

无

8.21 Radial Shading

8.21.1 功能描述

Radial Shading 模块是采用同心圆方式对镜头阴影现象进行矫正的模块。作为 Mesh Shading 模块功能上的补充，主要用来对鱼眼等特殊镜头进行镜头阴影矫正的功能。为了适配鱼眼镜头，当选择需要校正的成像范围（通过四个 OffCenter 参数调节）小于图像显示的范围时，会对成像范围外的图像进行裁剪。

8.21.2 API 参考

- [xmedia_isp_set_radial_shading_attr](#)：设置 radial shading 属性参数。
- [xmedia_isp_get_radial_shading_attr](#)：获取 radial shading 属性参数。
- [xmedia_isp_set_radial_shading_lut](#)：设置 radial shading LUT 属性参数。
- [xmedia_isp_get_radial_shading_lut](#)：获取 radial shading LUT 属性参数。

xmedia_isp_set_radial_shading_attr

【描述】

设置 radial shading 属性参数。

【语法】

```
xmedia_s32 xmedia_isp_set_radial_shading_attr(xmedia_u32 pipe, const
xmedia\_isp\_radial\_shading\_attr *ra_shading_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ra_shading_attr	ra_shading_attr 属性参数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意事项】

XM7206V11A 不支持

【举例】

无

【相关主题】

[xmedia_isp_get_radial_shading_attr](#)

xmedia_isp_get_radial_shading_attr

【描述】

获取 radial shading 属性参数。

【语法】

```
xmedia_s32 xmedia_isp_get_radial_shading_attr(xmedia_u32 pipe,
xmedia\_isp\_radial\_shading\_attr *ra_shading_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ra_shading_attr	ra_shading_attr 属性参数。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意事项】

XM7206V11A 不支持

【举例】

无

【相关主题】

[xmedia_isp_set_radial_shading_attr](#)

xmedia_isp_set_radial_shading_lut

【描述】

设置 radial shading 对应 LUT 参数。

【语法】

```
xmedia_s32 xmedia_isp_set_radial_shading_lut(xmedia_u32 pipe, const
xmedia\_isp\_radial\_shading\_lut\_attr *ra_shading_lut_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ra_shading_lut_attr	ra_shading_lut_attr 属性参数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意事项】

XM7206V11A 不支持

【举例】

无

【相关主题】

[xmedia_isp_get_radial_shading_lut](#)

xmedia_isp_get_radial_shading_lut

【描述】

获取 radial shading LUT 属性参数。

【语法】

```
xmedia_s32 xmedia_isp_get_radial_shading_lut(xmedia_u32 pipe,
xmedia\_isp\_radial\_shading\_lut\_attr *ra_shading_lut_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ra_shading_lut_attr	ra_shading_lut_attr 属性参数。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意事项】

XM7206V11A 不支持

【举例】

无

【相关主题】

[xmedia_isp_set_radial_shading_lut](#)

8.21.3 数据类型

- [xmedia_isp_radial_shading_attr](#): 定义 Radial Shading 属性参数。
- [xmedia_isp_radial_shading_gain_lut](#): 定义和传递 Radial Shading 不同光源及不同通道下的 Gain 值。
- [xmedia_isp_radial_shading_lut_attr](#): 定义 Radial Shading LUT 属性参数。

xmedia_isp_radial_shading_attr

【说明】

定义 Radial Shading 属性参数。

【定义】

```
typedef struct {  
    xmedia_bool enable;  
    xmedia_u8    iso_num;  
    xmedia_u32   iso_level[XMEDIA_ISP_ISO_MAX_COUNT];  
    xmedia_u16   radial_str[XMEDIA_ISP_ISO_MAX_COUNT];  
} xmedia_isp_radial_shading_attr;
```

【成员】

成员名称	描述
enable	Radial_shading 使能。
iso_num	ISO 数目。
iso_level	ISO 等级。
radial_str	镜头阴影矫正强度。

【注意事项】

XM7206V11A 不支持。

【相关数据类型及接口】

无

xmedia_isp_radial_shading_gain_lut

【说明】

定义和传递 Radial Shading 不同光源及不同通道下的 Gain 值。

【定义】

```
typedef struct {  
    xmedia_u16 wb_rgain;  
    xmedia_u16 wb_grgain;  
    xmedia_u16 wb_gbgain;  
    xmedia_u16 wb_bgain;  
    xmedia_u16 r_gain[XMEDIA_ISP_RLSC_POINTS];  
    xmedia_u16 gr_gain[XMEDIA_ISP_RLSC_POINTS];  
    xmedia_u16 gb_gain[XMEDIA_ISP_RLSC_POINTS];  
    xmedia_u16 b_gain[XMEDIA_ISP_RLSC_POINTS];  
}
```

```
} xmedia_isp_radial_shading_gain_lut;
```



说明

宏定义如下：

```
#define XMEDIA_ISP_RLSC_POINTS    129
```

【成员】

成员名称	描述
wb_rgain	表示该光源下的 R 通道 WB Gain 值。
wb_grgain	表示该光源下的 Gr 通道 WB Gain 值。
wb_gbgain	表示该光源下的 Gb 通道 WB Gain 值。
wb_bgain	表示该光源下的 B 通道 WB Gain 值。
r_gain	表示该光源下的 R 通道 Gain 值。
gr_gain	表示该光源下的 Gr 通道 Gain 值。
gb_gain	表示该光源下的 Gb 通道 Gain 值。
b_gain	表示该光源下的 B 通道 Gain 值。

【注意事项】

XM7206V11A 不支持。

【相关数据类型及接口】

[xmedia_isp_radial_shading_lut_attr](#)

xmedia_isp_radial_shading_lut_attr

【说明】

定义 Radial Shading LUT 属性参数。

【定义】

```
typedef struct {  
    xmedia\_video\_operation\_mode          light_mode;  
    xmedia_u16                          blend_ratio;  
    xmedia_u8                           light_type1;  
    xmedia_u8                           light_type2;  
    xmedia_u8                           radial_scale;
```

```

xmedia_u16          center_r_x;
xmedia_u16          center_r_y;
xmedia_u16          center_gr_x;
xmedia_u16          center_gr_y;
xmedia_u16          center_gb_x;
xmedia_u16          center_gb_y;
xmedia_u16          center_b_x;
xmedia_u16          center_b_y;
xmedia_u16          off_center_r;
xmedia_u16          off_center_gr;
xmedia_u16          off_center_gb;
xmedia_u16          off_center_b;
xmedia_isc_radial_shading_gain_lut r_isc_gain_lut[XMEDIA_ISP_RLSC_GAIN_LUT_NUM];
} xmedia_isc_radial_shading_lut_attr;

```



说明

宏定义如下：

```
#define XMEDIA_ISP_RLSC_GAIN_LUT_NUM 3
```

【成员】

成员名称	描述
light_mode	光源选择为自动模式还是手动模式。
blend_ratio	光源选择为手动模式下，光源 1 在融合时的权重值，默认为 256。
light_type1	光源 1 的选择为第 x 套 LUT 表，默认值 0。
light_type2	光源 2 的选择为第 x 套 LUT 表，默认值 0。
radial_scale	表示输入 LUT 的精度。
center_r_x	表示图像的光学中心 R 分量 x 轴坐标，默认取图像 1/2 的位置。
center_r_y	表示图像的光学中心 R 分量 y 轴坐标，默认取图像 1/2 的位置。
center_gr_x	表示图像的光学中心 Gr 分量 x 轴坐标，默认取图像 1/2 的位置。
center_gr_y	表示图像的光学中心 Gr 分量 y 轴坐标，默认取图像 1/2 的位置。
center_gb_x	表示图像的光学中心 Gb 分量 x 轴坐标，默认取图像 1/2 的位置。
center_gb_y	表示图像的光学中心 Gb 分量 y 轴坐标，默认取图像 1/2 的位置。
center_b_x	表示图像的光学中心 B 分量 x 轴坐标，默认取图像 1/2 的位置。

成员名称	描述
center_b_y	表示图像的光学中心 B 分量 y 轴坐标，默认取图像 1/2 的位置。
off_center_r	R 通道与校正范围半径相关的量，默认由中心点计算得到。
off_center_gr	Gr 通道与校正范围半径相关的量，默认由中心点计算得到。
off_center_gb	Gb 通道与校正范围半径相关的量，默认由中心点计算得到。
off_center_b	B 通道与校正范围半径相关的量，默认由中心点计算得到。
r_lsc_gain_lut	表示不同光源下的增益值（LUT）表，每种光源包含 4 个不同通道（R/Gr/Gb/B）的增益，每个通道用 129x16bit 的 LUT 表表示。

【注意事项】

XM7206V11A 不支持。

【相关数据类型及接口】

[xmedia_isp_radial_shading_gain_lut](#)

8.22 RGB-IR

8.22.1 功能描述

须知

- 目前市售大部分用于 IPC 产品形态的 RGB-IR 4x4 pattern 芯片波长在近红外区间（NIR，波长为 850nm）。如果需要使用 RGB-IR sensor 需要同时配合对应波长的双通滤光片使用。

芯片中所包含 RGB-IR 模块可用于处理排列满足 RGB-IR 4x4 pattern(以下简称为 RGB-IR4x4)的 RGB-IR sensor（RGB-IR 2x2 pattern 和 RGB-IR 4x4 pattern 的区别请参考图 8-9 和图 8-10）。通过本模块的开启，ISP 系统可识别 RGB-IR4x4 信号并处理为正确的颜色信号，同时可输出当前帧对应的 IR 分量图像供其他应用。

图8-9 传统 RGB-IR 2x2 pattern 排列

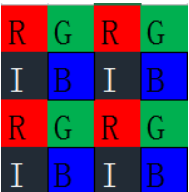
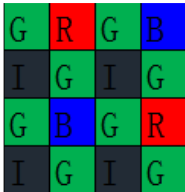


图8-10 RGB-IR 4x4 pattern 排列



8.22.2 API 参考

- `xmedia_isp_set_rgbir_attr`: 设定 RGBIR 属性。
- `xmedia_isp_get_rgbir_attr`: 获取 RGBIR 属性。

`xmedia_isp_set_rgbir_attr`

【描述】

设定 RGBIR 属性。

【语法】

```
xmedia_s32 xmedia_isp_set_rgbir_attr(xmedia_u32 pipe, const xmedia_isp_rgbir_attr *rgbir_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
rgbir_attr	rgbir_attr 属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：xmedia_isp.h
- 库文件：libxmedia_isp.a

【注意】

XM7206V11A 不支持。

【举例】

无。

【相关主题】

[xmedia_isp_get_rgbir_attr](#)

xmedia_isp_get_rgbir_attr

【描述】

获取 RGBIR 属性。

【语法】

```
xmedia_s32 xmedia_isp_get_rgbir_attr(xmedia_u32 pipe, xmedia_isp_rgbir_attr *rgbir_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
rgbir_attr	rgbir_attr 属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

XM7206V11A 不支持。

【举例】

无。

【相关主题】

[xmedia_isp_set_rgbir_attr](#)

8.22.3 数据类型

- [xmedia_isp_ir_cvtnat_mode](#): 定义 IR 转换矩阵模式。
- [xmedia_isp_irbayer_format](#): 定义 IR Bayer 格式。
- [xmedia_isp_rgbir_attr](#): 定义 RGBIR 转换模块属性。

xmedia_isp_ir_cvtnat_mode

【说明】

定义 IR 转换矩阵模式。

【定义】

```
typedef enum {  
    XMEDIA_ISP_IR_CVTMAT_MODE_NORMAL = 0,  
    XMEDIA_ISP_IR_CVTMAT_MODE_MONO   = 1,  
    XMEDIA_ISP_IR_CVTMAT_MODE_USER    = 2,  
} xmedia_isp_ir_cvtnat_mode;
```

【成员】

成员名称	描述
XMEDIA_ISP_IR_CVTMAT_MODE_NORMAL	枚举定义, RGBIR 模式选择, 彩色模式, 正常使用默认 CVTMAT 进行 IR 去除功能。
XMEDIA_ISP_IR_CVTMAT_MODE_MONO	枚举定义, RGBIR 模式选择, 夜晚红外模式下黑白图像。
XMEDIA_ISP_IR_CVTMAT_MODE_USER	枚举定义, RGBIR 模式选择, 用户自定义模式, 转换矩阵可自选。

【注意事项】

XM7206V11A 不支持

【相关数据类型及接口】

[xmedia_isp_rgbir_attr](#)

[xmedia_isp_irbayer_format](#)

【说明】

定义 IR Bayer 格式。

【定义】

```
typedef enum {
    XMEDIA_ISP_IRBAYER_FORMAT_GRGBI = 0,
    XMEDIA_ISP_IRBAYER_FORMAT_RGBGI = 1,
    XMEDIA_ISP_IRBAYER_FORMAT_GBGR = 2,
    XMEDIA_ISP_IRBAYER_FORMAT_BGRGI = 3,
    XMEDIA_ISP_IRBAYER_FORMAT_IGRGB = 4,
    XMEDIA_ISP_IRBAYER_FORMAT_IRGBG = 5,
    XMEDIA_ISP_IRBAYER_FORMAT_IBGRG = 6,
    XMEDIA_ISP_IRBAYER_FORMAT_IGBGR = 7,
    XMEDIA_ISP_IRBAYER_FORMAT_MAX
} xmedia_isp_irbayer_format;;
```

【成员】

成员名称	描述
XMEDIA_ISP_IRBAYER_FORMAT_GRGBI	GRGBI 排列方式。

成员名称	描述
XMEDIA_ISP_IRBAYER_FORMAT_RGBGI	RGBGI 排列方式。
XMEDIA_ISP_IRBAYER_FORMAT_GBGR	GBGR 排列方式。
XMEDIA_ISP_IRBAYER_FORMAT_BGRGI	BGRGI 排列方式。
XMEDIA_ISP_IRBAYER_FORMAT_IGRGB	IGRGB 排列方式。
XMEDIA_ISP_IRBAYER_FORMAT_IGRBG	IGRBG 排列方式。
XMEDIA_ISP_IRBAYER_FORMAT_IBGRG	IBGRG 排列方式。
XMEDIA_ISP_IRBAYER_FORMAT_IGBGR	IGBGR 排列方式。

【注意事项】

XM7206V11A 不支持

【相关数据类型及接口】

xmedia_isp_rgbir_attr

xmedia_isp_rgbir_attr

【说明】

定义 RGBIR 转换模块属性。

【定义】

```
typedef struct {
    xmedia_bool          enable;
    xmedia_isp_irbayer_format in_pattern;
    xmedia_video_bayer_format out_pattern;
    xmedia_u16            exp_ctrl[XMEDIA_ISP_EXP_CTRL_NUM];
    xmedia_u16            gain[XMEDIA_ISP_EXP_CTRL_NUM];
    xmedia_isp_ir_cvtmat_mode ir_cvt_mat_mode;
    xmedia_s16            cvt_matrix[XMEDIA_ISP_CVTMAT_NUM];
} xmedia_isp_rgbir_attr;
```



说明

宏定义如下：

```
#define XMEDIA_ISP_EXP_CTRL_NUM 2
#define XMEDIA_ISP_CVTMAT_NUM 12
```

【成员】

成员名称	描述	取值范围	默认值
enable	RGBIR 模块使能模块。	[0,1]	0
in_pattern	RGBIR 模块输入 pattern 选择	[0,7]	-
out_pattern	RGBIR 模块输出格式选择，与普通 Bayer Pattern 一致。	[0,3]	-
exp_ctrl	过曝处理模块相关，曝光亮度区间确定参数。	[0,2047]	exp_ctrl[0] = 1500 exp_ctrl[1] = 300
gain	过曝处理模块相关，在通过 exp_ctrl 确定了区间后，使用本参数进行过曝处理。	[0,511]	gain[0] = 256 gain[1] = 256
ir_cvt_mat_mode	RGBIR 转换矩阵功能选择，分 Normal 模式，Mono 模式和 User 模式。	[0,2]	0
cvt_matrix	RGBIR 转换矩阵参数配置。Normal 模式和 Mono 模式下使用系统默认参数进行配置，User 模式下可配置。	[-32768,32767]	-

【注意事项】

XM7206V11A 不支持

【相关数据类型及接口】

[xmedia_isp_set_rgbir_attr](#)

[xmedia_isp_get_rgbir_attr](#)

8.23 Expander

8.23.1 功能描述

部分 sensor 内部会做多帧曝光的融合，融合后数据位宽会增大，导致输出成本成大；为减少输出的成本，sensor 内部会做数据分段压缩，将数据压缩到一个比较小的位宽；在 ISP 中为了将数据还原，需要将 sensor 内部压缩的数据，进行解压缩。

8.23.2 API 参考

- `xmedia_isp_set_expander_attr`: 设定 Expander 属性。
- `xmedia_isp_get_expander_attr`: 获取 Expander 属性。

`xmedia_isp_set_expander_attr`

【描述】

设定 Expander 属性。

【语法】

```
xmedia_s32 xmedia_isp_set_expander_attr(xmedia_u32 pipe, const xmedia_isp_expander_attr *  
expander_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
expander_attr	expander_attr 属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: `xmedia_isp.h`
- 库文件: `libxmedia_isp.a`

【注意】

XM7606V13 使用注意:

Built in wdr 模式下，不能关闭 expander 算法；

非 Built in wdr 模式下，不能打开 expander 算法。

【举例】

无。

【相关主题】

[xmedia_isp_get_expander_attr](#)

xmedia_isp_get_expander_attr

【描述】

获取 EXPANDER 属性。

【语法】

```
xmedia_s32 xmedia_isp_get_expander_attr(xmedia_u32 pipe, xmedia\_isp\_expander\_attr
*expander_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
expander_attr	expander_attr 属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无。

【举例】

无。

【相关主题】

[xmedia_isp_set_expander_attr](#)

8.23.3 数据类型

- [xmedia_isp_expander_attr](#): 定义 Expander 模块属性。

xmedia_isp_expander_attr

【说明】

定义 EXPANDER 转换模块属性。

【定义】

```
typedef struct {  
    xmedia_bool        enable;  
    xmedia_u8          bit_depth_in;  
    xmedia_u8          bit_depth_out;  
    xmedia_u16         knee_point_num;  
    xmedia_video_point knee_point_coord[XMEDIA_ISP_EXPANDER_KNEE_POINT_NUM];  
} xmedia_isp_expander_attr;
```



说明

宏定义如下:

```
#define XMEDIA_ISP_EXPANDER_KNEE_POINT_NUM 256
```

【成员】

成员名称	描述	取值范围
enable	expander 模块使能模块。	[0,1]
bit_depth_in	输入数据位宽，根据 sensor 实际 raw 位宽设置	[0xC,0x14]
bit_depth_out	输出数据位宽。 XM7606V13 固定为 0x14， XM7206V11A 固定为 0x11	
knee_point_num	拐点坐标的数目。 XM7606V13 取值范围：[1,256] XM7206V11A 取值范围：[1,128]	
knee_point_coord	解压的拐点，包括横纵坐标，必须单调递增。 XM7606V13 - X 取值范围：[0x0,0x100] - Y 取值范围：[0x0, 0x100000] XM7206V11A - X 取值范围：[0x0,0x80] - Y 取值范围：[0x0, 0x20000]	

【注意事项】

在 sensor 手册中会给出 sensor 在 sensor built-in 模式下内部压缩时使用的拐点配置，需要将这几个拐点配置按照对应的原则进行转换，然后配置到 knee_point_coord 中即可。

XM7606V13 转换原则如下：

- knee_point_coord 的横坐标 x 需要根据 sensor 压缩曲线转换到[0,256]之间 (8bit) ，例如 sensor 压缩输出的后有效数据位宽是 12bit，则需要将 sensor 压缩曲线拐点的纵坐标右移 4bit 得到 knee_point_coord 的横坐标 x；
- knee_point_coord 的纵坐标 y 需要根据 sensor 压缩曲线转换到[0,1048576]之间 (20bit) ，例如 sensor 合成有效数据未压缩之前有效位宽是 16bit，则需要将 sensor 压缩曲线的拐点的横坐标左移 4bit，得到 knee_point_coord 的纵坐标 y。

- 不必配置(0,0)的拐点坐标，代码内部在进行插值时，会自动将(0,0)作为第一个拐点坐标。高频次的动态修改 expander 的拐点配置，可能会导致同一幅图像使用不同的解压缩曲线

XM7206V11A 转换原则如下：

- knee_point_coord 的横坐标 x 需要根据 sensor 压缩曲线转换到[0,128]之间 (7bit)，例如 sensor 压缩输出的后有效数据位宽是 12bit，则需要将 sensor 压缩曲线拐点的纵坐标右移 5bit 得到 knee_point_coord 的横坐标 x；
- knee_point_coord 的纵坐标 y 需要根据 sensor 压缩曲线转换到[0,131071]之间 (17bit)，例如 sensor 合成有效数据未压缩之前有效位宽是 16bit，则需要将 sensor 压缩曲线的拐点的横坐标左移 1bit，得到 knee_point_coord 的纵坐标 y。
- 不必配置(0,0)的拐点坐标，代码内部在进行插值时，会自动将(0,0)作为第一个拐点坐标。高频次的动态修改 expander 的拐点配置，可能会导致同一幅图像使用不同的解压缩曲线

expander 拐点会影响 built-in 模式下 BE 黑电平的配置，动态调节 expander 拐点，图像可能会有颜色的跳变。

【相关数据类型及接口】

[xmedia_isp_set_expander_attr](#)

[xmedia_isp_get_expander_attr](#)

8.24 STNR

8.24.1 功能描述

STNR 模块整合 YNR(亮度空间域去噪)、TNR(时间域去噪)、CNR(色度空间域去噪)三个 YUV 域的主要降噪算法模块，进行 YUV 域的去噪处理。

8.24.2 API 参考

- [xmedia_isp_set_stnr_attr](#)：设定 Stnr 属性。
- [xmedia_isp_get_stnr_attr](#)：获取 Stnr 属性。

[xmedia_isp_set_stnr_attr](#)

【描述】

设定 Stnr 属性。

【语法】

```
xmedia_s32 xmedia_isp_set_stnr_attr(xmedia_u32 pipe, const xmedia_isp_stnr_attr *stnr_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
stnr_attr	stnr_attr 属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

XM7606V13 暂未支持。

【举例】

无。

【相关主题】

[xmedia_isp_get_stnr_attr](#)

xmedia_isp_get_stnr_attr

【描述】

获取 Stnr 属性。

【语法】

xmedia_s32 xmedia_isp_get_stnr_attr(xmedia_u32 pipe, [xmedia_isp_stnr_attr](#) *stnr_attr);

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
stnr_attr	stnr_attr 属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

XM7606V13 暂未支持。

【举例】

无。

【相关主题】

[xmedia_isp_set_stnr_attr](#)

8.24.3 数据类型

- [xmedia_isp_stnr_attr](#): 定义 STNR 模块属性。
- [xmedia_isp_stnr_ynr_param](#): 定义 ynr 属性。
- [xmedia_isp_stnr_ynr_noise_str_mode](#): 定义 ynr 去噪基础强度是否参考运动区的模式属性。
- [xmedia_isp_stnr_ynr_grad_win_size](#): 定义 ynr 计算纹理所用窗口大小属性。
- [xmedia_isp_stnr_ynr_var_mode](#): 定义 ynr 的方差是否结合运动信息模式选择属性。

- `xmedia_isp_stnr_ynr_select_centerluma_mode`: 定义 ynr 中心点模式选择属性。
- `xmedia_isp_stnr_ynr_addback_scl_mode`: 定义 ynr 的噪声回叠是否考虑运动信息模式选择属性。
- `xmedia_isp_stnr_ynr_param_manual`: 定义 ynr 手动参数属性。
- `xmedia_isp_stnr_ynr_param_auto`: 定义 ynr 自动参数属性。
- `xmedia_isp_stnr_tnr_param`: 定义 tnr 属性。
- `xmedia_isp_stnr_tnr_param_manual`: 定义 tnr 手动参数属性。
- `xmedia_isp_stnr_tnr_param_auto`: 定义 tnr 自动属性。
- `xmedia_isp_stnr_cnr_param`: 定义 cnr 属性。
- `xmedia_isp_stnr_cnr_param_manual`: 定义 cnr 手动参数属性。
- `xmedia_isp_stnr_cnr_param_auto`: 定义 cnr 自动参数属性。

xmedia_isp_stnr_attr

【说明】

定义 STNR 模块属性。

【定义】

```
typedef struct {
    xmedia_bool stnr_enable;
    xmedia_u8   stnr_mode;
    xmedia_bool ynr_enable;
    xmedia_bool cnr_enable;
    xmedia_bool tnr_enable;
    xmedia_bool tnr_mc_enable;
    xmedia_isp_stnr_ynr_param ynr;
    xmedia_isp_stnr_tnr_param tnr;
    xmedia_isp_stnr_cnr_param cnr;
} xmedia_isp_stnr_attr;
```

【成员】

成员名称	描述
stnr_enable	stnr 总开关。 取值范围: [0,1]。
stnr_mode	选择 YNR/TNR/YSH 顺序的模式, XM7606V13 不支

成员名称	描述
	持。 取值范围：[0,2]。 0：YSH->YNR->TNR(B3D 使能时，必须为 0)。 1：TNR->YSH->YNR(仅 B3Doff 运行，暗处对比度较 2 稍强)。 2：YSH->TNR->YNR(仅 B3Doff 运行，暗处亮度较 1 稍亮，存在可能偏红现象)。
ynr_enable	亮度域空域去噪使能。 取值范围：[0,1]。
cnr_enable	色度域空域去噪使能。 取值范围：[0,1]。
tnr_enable	时域去噪使能。 取值范围：[0,1]。
tnr_mc_enable	时域运动补偿使能。 取值范围：[0,1]。 XM7606V13 和 XM7206V11A 均不支持
ynr	亮度域空域去噪参数。
tnr	时域去噪参数。
cnr	色度域空域去噪参数。

【注意事项】

无。

【相关数据类型及接口】

[xmedia_isp_set_stnr_attr](#)

[xmedia_isp_get_stnr_attr](#)

[xmedia_isp_stnr_ynr_param](#)

【说明】

定义 ynr 属性。

【定义】

```
typedef struct {  
    xmedia_isp_stnr_ynr_noise_str_mode          denoise_str_mode;  
    xmedia_isp_stnr_ynr_grad_win_size           grad_win_size;  
    xmedia_isp_stnr_ynr_var_mode                var_mode;  
    xmedia_isp_stnr_ynr_select_centerluma_mode  select_centerluma_mode;  
    xmedia_bool                                 nr_addback_en;  
    xmedia_isp_stnr_ynr_addback_scl_mode        ynr_addback_scl_mode;  
    xmedia_video_operation_mode                 op_type;  
    xmedia_isp_stnr_ynr_param_manual            manual_attr;  
    xmedia_isp_stnr_ynr_param_auto              auto_attr;  
} xmedia_isp_stnr_ynr_param;
```

【成员】

成员名称	描述
denoise_str_mode;	去噪基础强度是否参考运动区模式选择。 取值范围：[0,1]。XM7606V13 不支持。
grad_win_size	计算梯度的窗口大小模式选择。 取值范围：5,7,9。XM7606V13 不支持。
var_mode	edge 图是否参考运动区模式选择。 取值范围：[0,1]。XM7606V13 不支持。
select_centerluma_mode	中心点模式选择。 取值范围：[0,1]。XM7606V13 不支持。
nr_addback_en	噪声回叠开关。 取值范围：[0,1]。XM7606V13 不支持。
ynr_addback_scl_mode	噪声回叠强度是否参考运动区模式选择。 取值范围：[0,1]。XM7606V13 不支持。
op_type	ynr 操作模式，0：自动模式；1：手动模式。
manual_attr	ynr 手动参数结构。
auto_attr	ynr 自动参数结构。

【注意事项】

无。

【相关数据类型及接口】

[xmedia_isp_set_stnr_attr](#)

[xmedia_isp_get_stnr_attr](#)

xmedia_isp_stnr_ynr_noise_str_mode

【说明】

定义 ynr 去噪基础强度是否参考运动区的模式属性。

【定义】

```
typedef enum {  
    XMEDIA_ISP_STNR_YNR_DENOISE_STR_DEFAULT = 0,  
    XMEDIA_ISP_STNR_YNR_DENOISE_STR_MOV      = 1,  
    XMEDIA_ISP_STNR_YNR_DENOISE_STR_MAX  
} xmedia_isp_stnr_ynr_noise_str_mode;
```

【成员】

成员名称	描述
XMEDIA_ISP_STNR_YNR_DENOISE_STR_DEFAULT	去噪基础强度默认不参考运动区
XMEDIA_ISP_STNR_YNR_DENOISE_STR_MOV	去噪基础强度参考运动区

【注意事项】

XM7606V13 不支持。

【相关数据类型及接口】

[xmedia_isp_set_stnr_attr](#)

[xmedia_isp_get_stnr_attr](#)

xmedia_isp_stnr_ynr_grad_win_size

【说明】

定义 ynr 计算纹理所用窗口大小属性。

【定义】

```
typedef enum {  
    XMEDIA_ISP_STNR_YNR_GRAD_WIN_SML = 5,  
    XMEDIA_ISP_STNR_YNR_GRAD_WIN_MED = 7,  
    XMEDIA_ISP_STNR_YNR_GRAD_WIN_LG  = 9,  
    XMEDIA_ISP_STNR_YNR_GRAD_WIN_MAX  
} xmedia_isp_stnr_ynr_grad_win_size;
```

【成员】

成员名称	描述
XMEDIA_ISP_STNR_YNR_GRAD_WIN_SML	小窗口计算梯度
XMEDIA_ISP_STNR_YNR_GRAD_WIN_MED	中等窗口计算梯度
XMEDIA_ISP_STNR_YNR_GRAD_WIN_LG	大窗口计算梯度

【注意事项】

XM7606V13 不支持。

【相关数据类型及接口】

[xmedia_isp_set_stnr_attr](#)

[xmedia_isp_get_stnr_attr](#)

xmedia_isp_stnr_ynr_var_mode

【说明】

定义 ynr 的方差是否结合运动信息模式选择属性。

【定义】

```
typedef enum {
    XMEDIA_ISP_STNR_YNR_VAR_MODE_DEFAULT = 0,
    XMEDIA_ISP_STNR_YNR_VAR_MODE_MOV      = 1,
    XMEDIA_ISP_STNR_YNR_VAR_MODE_MAX
} xmedia_isp_stnr_ynr_var_mode;
```

【成员】

成员名称	描述
XMEDIA_ISP_STNR_YNR_VAR_MODE_DEFAULT	默认不参考运动区算边缘
XMEDIA_ISP_STNR_YNR_VAR_MODE_MOV	参考运动区算边缘

【注意事项】

XM7606V13 不支持。

【相关数据类型及接口】

[xmedia_isp_set_stnr_attr](#)

[xmedia_isp_get_stnr_attr](#)

xmedia_isp_stnr_ynr_select_centerluma_mode

【说明】

定义 ynr 中心点模式选择属性。

【定义】

```
typedef enum {
    XMEDIA_ISP_STNR_YNR_SELECT_CENTERLUMA_NR = 0,
    XMEDIA_ISP_STNR_YNR_SELECT_CENTERLUMA_ORI = 1,
    XMEDIA_ISP_STNR_YNR_SELECT_CENTERLUMA_MAX
} xmedia_isp_stnr_ynr_select_centerluma_mode;
```

【成员】

成员名称	描述
XMEDIA_ISP_STNR_YNR_SELECT_CENTERLUMA_NR	经过滤波后的结果作为中心点像素
XMEDIA_ISP_STNR_YNR_SELECT_CENTERLUMA_ORI	原始像素作为中心点像素

【注意事项】

XM7606V13 不支持。

【相关数据类型及接口】

[xmedia_isp_set_stnr_attr](#)

[xmedia_isp_get_stnr_attr](#)

xmedia_isp_stnr_ynr_addback_scl_mode

【说明】

定义 ynr 的噪声回叠是否考虑运动信息模式选择属性。

【定义】

```
typedef enum {
```

```

XMEDIA_ISP_STNR_YNR_ADDBACK_SCL_DEFAULT = 0,
XMEDIA_ISP_STNR_YNR_ADDBACK_SCL_MOV      = 1,
XMEDIA_ISP_STNR_YNR_ADDBACK_SCL_MAX
} xmedia_isp_stnr_ynr_addback_scl_mode;

```

【成员】

成员名称	描述
XMEDIA_ISP_STNR_YNR_ADDBACK_SCL_DEFAULT	噪声回叠强度不参考运动区
XMEDIA_ISP_STNR_YNR_ADDBACK_SCL_MOV	噪声回叠强度参考运动区

【注意事项】

XM7606V13 不支持。

【相关数据类型及接口】

[xmedia_isp_set_stnr_attr](#)

[xmedia_isp_get_stnr_attr](#)

[xmedia_isp_stnr_ynr_param_manual](#)

【说明】

定义 ynr 手动参数属性。

【定义】

```

typedef struct {
    xmedia_u8  edge_sens;
    xmedia_u8  corner_sens;
    xmedia_u8  struct_sens;
    xmedia_u8  graininess_sens;
    xmedia_u8  v_denoise_str;
    xmedia_u8  s_denoise_str;
    xmedia_u16 min_denoise_str;
    xmedia_u8  sigma;
    xmedia_u8  s_luma_thd_low;
    xmedia_u8  s_luma_thd_high;
    xmedia_u8  s_luma_str_low;
    xmedia_u8  s_luma_str_high;
    xmedia_u8  s_mot_thd_low;

```

```

xmedia_u8 s_mot_thd_high;
xmedia_u8 s_mot_str_low;
xmedia_u8 s_mot_str_high;
xmedia_u8 v_luma_thd_low;
xmedia_u8 v_luma_thd_high;
xmedia_u8 v_luma_str_low;
xmedia_u8 v_luma_str_high;
xmedia_u8 v_mot_thd_low;
xmedia_u8 v_mot_thd_high;
xmedia_u8 v_mot_str_low;
xmedia_u8 v_mot_str_high;
xmedia_u8 text_thd[XMEDIA_ISP_3DNR_THD_MAX_COUNT];
xmedia_u8 text_str[XMEDIA_ISP_3DNR_STR_MAX_COUNT];
xmedia_u8 luma_varscale_thd_low;
xmedia_u8 luma_varscale_thd_high;
xmedia_u16 luma_varscale_str_low;
xmedia_u16 luma_varscale_str_high;
xmedia_u16 var_scale;
xmedia_u8 blend_var_thd_low;
xmedia_u8 blend_var_thd_high;
xmedia_u16 blend_var_low_wgt;
xmedia_u16 blend_var_high_wgt;
xmedia_u8 blend_luma_thd_low;
xmedia_u8 blend_luma_thd_high;
xmedia_u16 blend_luma_low_wgt;
xmedia_u16 blend_luma_high_wgt;
xmedia_u8 blend_mot_thd_low;
xmedia_u8 blend_mot_thd_high;
xmedia_u16 blend_mot_low_wgt;
xmedia_u16 blend_mot_high_wgt;
xmedia_u16 nr_addback_str;
xmedia_u8 blend_nr_center_var_thd_low;
xmedia_u8 blend_nr_center_var_thd_high;
xmedia_u16 blend_nr_center_var_low_wgt;
xmedia_u16 blend_nr_center_var_high_wgt;
xmedia_u8 addback_text_thd[XMEDIA_ISP_3DNR_THD_MAX_COUNT];
xmedia_u16 addback_text_str[XMEDIA_ISP_3DNR_STR_MAX_COUNT];
xmedia_u8 addback_luma_thd[XMEDIA_ISP_3DNR_THD_MAX_COUNT];
xmedia_u16 addback_luma_str[XMEDIA_ISP_3DNR_STR_MAX_COUNT];
xmedia_u8 addback_text_luma_blend_mot_thd_low;
xmedia_u16 addback_text_luma_blend_mot_thd_high;
xmedia_u8 addback_mot_thd[XMEDIA_ISP_3DNR_THD_MAX_COUNT];
xmedia_u16 addback_mot_weak_str;

```

```
xmedia_u16 addback_mot_strong_str;
} xmedia_isp_stnr_ynr_param_manual;
```



说明

宏定义如下:

```
#define XMEDIA_ISP_3DNR_THD_MAX_COUNT 4
#define XMEDIA_ISP_3DNR_STR_MAX_COUNT 3
```

【成员】

成员名称	描述
edge_sens	边缘区敏感指数, 取值范围[0,255]。
corner_sens	角点区敏感指数, 取值范围[0,255]。
struct_sens	角点区敏感指数, 取值范围[0,255]。
graininess_sens	颗粒性敏感指数, 取值范围[0,255]。
v_denoise_str	控制值域核去噪强度, 取值范围[0,255]。
s_denoise_str	控制空间核去噪强度 , 取值范围[0,255]。 XM7206V111A 不支持。
min_denoise_str	最小去噪强度, 取值范围[0,511]。 XM7606V13 不支持。
sigma	滤波参数 sigma , 取值范围[0,127]。 XM7606V13 不支持。
s_luma_thd_low	根据运动信息调整空间核的去噪强度。
s_luma_thd_high	

成员名称	描述
s_luma_str_low	如果运动强度小于 s_mot_thd_low 判定静止区，则用 s_mot_str_low 调整； 如果运动强度大于 s_mot_thd_high 判定运动区，则用 s_mot_str_high 调整；如果运动强度介于两个阈值之间，则采用线性插值的方法得到调整量。 取值范围均为[0,255]。 XM7206V111A 不支持。
s_luma_str_high	
s_mot_thd_low	根据亮度调整空间核的去噪强度。 如果图像亮度小于 s_luma_thd_low，则用 s_luma_str_low 调整； 如果图像亮度大于 s_luma_thd_high，则用 s_luma_str_high 调整； 如果图像亮度介于两个阈值之间，则采用线性插值的方法得到调整量。 取值范围均为[0,255]。 XM7206V111A 不支持。
s_mot_thd_high	
s_mot_str_low	
s_mot_str_high	
v_luma_thd_low	根据亮度调整值域核的去噪强度。 如果图像亮度小于 v_luma_thd_low，则用 v_luma_str_low 调整； 如果图像亮度大于 v_luma_thd_high，则用 v_luma_str_high 调整； 如果图像亮度介于两个阈值之间，则采用线性插值的方法得到调整量。 取值范围均为[0,255]。
v_luma_thd_high	
v_luma_str_low	
v_luma_str_high	
v_mot_thd_low	根据运动信息调整值域核的去噪强度。 如果运动强度小于 v_mot_thd_low 判定静止区，则用 v_mot_str_low 调整； 如果运动强度大于 v_mot_thd_high 判定运动区，则用 v_mot_str_high 调整； 如果运动强度介于两个阈值之间，则采用线性插值的方法得到调整量。 取值范围均为[0,255]。
v_mot_thd_high	
v_mot_str_low	
v_mot_str_high	
text_thd[XMEDIA_ISP_3]	

成员名称	描述
DNR_THD_MAX_COUNT]	根据纹理信息调整去噪强度。
text_str[XMEDIA_ISP_3 DNR_STR_MAX_COUNT]	如果小于 text_thd[0], 判定为平坦区, 用 text_str[0]强度; 如果大于 text_thd[1]且小于 text_thd[2], 判定为弱纹理区, 用 text_str[1]强度; 如果大于 text_thd[3], 判定为强边区, 用 text_str[2]强度。 如果介于 text_thd[0]和 text_thd[1]两个阈值之间, 则采用线性插值的方法得到调整量。 如果介于 text_thd[2]和 text_thd[3]两个阈值之间, 则采用线性插值的方法得到调整量。 取值范围均为[0,255]。
luma_varscale_thd_low	根据亮度调整方差大小。如果图像亮度小于 luma_varscale_thd_low, 则用 luma_varscale_str_low 调整; 如果图像亮度大于 luma_varscale_thd_high, 则用 luma_varscale_str_high 调整; 如果图像亮度介于两个阈值之间, 则采用线性插值的方法得到调整量。 XM7606V13 不支持。 luma_varscale_thd_low、luma_varscale_thd_high 取值范围为[0,255]; luma_varscale_str_low、luma_varscale_str_high 取值范围为[0,256]。
luma_varscale_thd_high	
luma_varscale_str_low	
luma_varscale_str_high	
var_scale	对方差结果进行收缩, 取值范围[0,1023]。 XM7606V13 不支持。
blend_var_thd_low	根据方差大小调整两种滤波方式融合比例。如果方差小于 blend_var_thd_low, 则用 YNR_ blend_var_low_wgt 调整; 如果方差大于 blend_var_thd_high, 则用 blend_var_high_wgt 调整; 如果方差介于两个阈值之间, 则采用线性插值的方法得到调整量。 XM7606V13 不支持。
blend_var_thd_high	
blend_var_low_wgt	
blend_var_high_wgt	

成员名称	描述
	blend_var_thd_low、blend_var_thd_high 取值范围为 [0,255]; blend_var_low_wgt、blend_var_high_wgt 取值范围为 [0,256]。
blend_luma_thd_low	根据亮度大小调整两种滤波方式融合比例。如果亮度小于 blend_luma_thd_low, 则用 YNR_ blend_luma_low_wgt 调整; 如果亮度大于 blend_luma_thd_high, 则用 blend_luma_high_wgt 调整; 如果亮度介于两个阈值之间, 则采用线性插值的方法得到调整量。XM7606V13 不支持。 blend_luma_thd_low、blend_luma_thd_high 取值范围为 [0,255]; blend_luma_low_wgt、blend_luma_high_wgt 取值范围为 [0,256]。
blend_luma_thd_high	
blend_luma_low_wgt	
blend_luma_high_wgt	
blend_mot_thd_low	根据运动大小调整两种滤波方式融合比例。如果运动小于 blend_mot_thd_low, 则用 blend_mot_low_wgt 调整; 如果运动大于 blend_mot_thd_high, 则用 blend_mot_high_wgt 调整; 如果运动介于两个阈值之间, 则采用线性插值的方法得到调整量。 blend_mot_thd_low、blend_mot_thd_high 取值范围为 [0,255]; blend_mot_low_wgt、blend_mot_high_wgt 取值范围为 [0,256]。 XM7606V13 不支持。
blend_mot_thd_high	
blend_mot_low_wgt	
blend_mot_high_wgt	
nr_addback_str	噪声回叠强度。 取值范围为 [0,256]。 XM7606V13 不支持。
blend_nr_center_var_thd_low	在噪声回叠开启情况下, 根据方差大小调整中心亮度和高斯滤波的融合比例。如果方差小于 blend_nr_center_var_thd_low, 则用 blend_nr_center_var_low_wgt 调整; 如果方差大于
blend_nr_center_var_thd_high	
blend_nr_center_var_lo	

成员名称	描述
w_wgt	blend_nr_center_var_thd_high, 则用 blend_nr_center_var_high_wgt 调整; 如果方差介于 两个阈值之间, 则采用线性插值的方法得到调整量。 XM7606V13 不支持。 blend_nr_center_var_thd_low、 blend_nr_center_var_thd_high 取值范围为[0,255]; blend_nr_center_var_low_wgt、 blend_nr_center_var_high_wgt 取值范围为[0,256]。
blend_nr_center_var_high_wgt	
addback_text_thd[XMEDIA_ISP_3DNR_THD_MAX_COUNT]	在噪声回叠开启情况下, 根据纹理信息调整回叠强度。如果小于 addback_text_thd[0], 用 addback_text_str[0]强度; 如果大于 addback_text_thd[1]且小于 addback_text_thd[2], 用 addback_text_str[1]强度; 如果大于 addback_text_thd[3], 用 addback_text_str[2]强度。 如果介于 addback_text_thd[0]和 addback_text_thd[1]两个阈值之间, 则采用线性插值 的方法得到调整量。 如果介于 addback_text_thd[2]和 addback_text_thd[3] 两个阈值之间, 则采用线性插值的方法得到调整量。 addback_text_thd 取值范围为[0,255], addback_text_str 取值范围为[0,511]。 XM7606V13 不支持。
addback_text_str[XMEDIA_ISP_3DNR_STR_MAX_COUNT]	
addback_luma_thd[XMEDIA_ISP_3DNR_THD_MAX_COUNT]	在噪声回叠开启情况下, 根据亮度调整回叠强度。如 果小于 addback_luma_thd[0], 用 addback_luma_str[0]强度; 如果大于 addback_luma_thd[1]且小于 addback_luma_thd[2], 用 addback_luma_str[1]强度; 如果大于 addback_luma_thd[3], 用 addback_luma_str[2]强 度。 如果介于 addback_luma_thd[0]和 addback_luma_thd[1]两个阈值之间, 则采用线性插值 的方法得到调整量。
addback_luma_str[XMEDIA_ISP_3DNR_STR_MAX_COUNT]	

成员名称	描述
	如果介于 addback_luma_thd[2]和 addback_luma_thd[3]两个阈值之间，则采用线性插值的方法得到调整量。XM7606V13 不支持。 addback_luma_thd 取值范围为[0,255]; addback_luma_str 取值范围为[0,511]。
addback_text_luma_blend_mot_thd_low	在噪声回叠开启情况下，根据运动信息融合用纹理信息调整前和调整后回叠强度。 addback_text_luma_blend_mot_thd_low 取值范围为 [0,255]。 addback_text_luma_blend_mot_thd_high 取值范围为 [0,256]。 XM7606V13 不支持。
addback_text_luma_blend_mot_thd_high	
addback_mot_thd[XMEDIA_ISP_3DNR_THD_MAX_COUNT]	在噪声回叠开启情况下，根据运动信息调整回叠强度。如果小于 addback_mot_thd[0]，用 256 强度；如果大于 addback_mot_thd[1]且小于 addback_mot_thd[2]，用 addback_mot_weak_str 强度；如果大于 addback_mot_thd[3]，用 addback_mot_strong_str 强度。 如果介于 addback_mot_thd[0]和 addback_mot_thd[1]两个阈值之间，则采用线性插值的方法得到调整量。 如果介于 addback_mot_thd[2]和 addback_mot_thd[3]两个阈值之间，则采用线性插值的方法得到调整量。 XM7606V13 不支持。 addback_mot_thd 取值范围为[0,255]。 addback_mot_weak_str 取值范围为[0,511]。 addback_mot_strong_str 取值范围为[0,256]。
addback_mot_weak_str	
addback_mot_strong_str	

【注意事项】

无。

【相关数据类型及接口】

[xmedia_isp_set_stnr_attr](#)

[xmedia_isp_get_stnr_attr](#)

xmedia_isp_stnr_ynr_param_auto

【说明】

定义 ynr 自动参数属性。

【定义】

```
typedef struct {
    xmedia_u32 iso_num;
    xmedia_u32 iso_level[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  edge_sens[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  corner_sens[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  struct_sens[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  graininess_sens[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  v_denoise_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  s_denoise_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16 min_denoise_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  sigma[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  s_luma_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  s_luma_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  s_luma_str_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  s_luma_str_high[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  s_mot_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  s_mot_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  s_mot_str_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  s_mot_str_high[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  v_luma_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  v_luma_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  v_luma_str_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  v_luma_str_high[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  v_mot_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  v_mot_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  v_mot_str_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  v_mot_str_high[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8
    text_thd[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_3DNR_THD_MAX_COUNT];
    xmedia_u8
    text_str[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_3DNR_STR_MAX_COUNT];
    xmedia_u8  luma_varscale_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  luma_varscale_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16 luma_varscale_str_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16 luma_varscale_str_high[XMEDIA_ISP_ISO_MAX_COUNT];
}
```

```

xmedia_u16 var_scale[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8  blend_var_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8  blend_var_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 blend_var_low_wgt[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 blend_var_high_wgt[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8  blend_luma_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8  blend_luma_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 blend_luma_low_wgt[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 blend_luma_high_wgt[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8  blend_mot_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8  blend_mot_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 blend_mot_low_wgt[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 blend_mot_high_wgt[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 nr_addback_str[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8  blend_nr_center_var_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8  blend_nr_center_var_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 blend_nr_center_var_low_wgt[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 blend_nr_center_var_high_wgt[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8
addback_text_thd[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_3DNR_THD_MAX_COUNT];
xmedia_u16
addback_text_str[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_3DNR_STR_MAX_COUNT];
xmedia_u8
addback_luma_thd[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_3DNR_THD_MAX_COUNT];
xmedia_u16
addback_luma_str[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_3DNR_STR_MAX_COUNT];
xmedia_u8  addback_text_luma_blend_mot_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 addback_text_luma_blend_mot_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8
addback_mot_thd[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_3DNR_THD_MAX_COUNT];
xmedia_u16 addback_mot_weak_str[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 addback_mot_strong_str[XMEDIA_ISP_ISO_MAX_COUNT];
} xmedia_isp_stnr_ynr_param_auto;

```

【成员】

成员名称	描述	取值范围
iso_num	ISO 等级	
iso_level[XMEDIA_ISP_ISO_MAX_COUNT]	ISO 数量	
edge_sens[XMEDIA_ISP_ISO_MAX_COUNT]	边缘区敏感指数	[0,255]

成员名称	描述	取值范围
corner_sens[XMEDIA_ISP_ISO_MAX_COUNT]	角点区敏感指数	[0,255]
struct_sens[XMEDIA_ISP_ISO_MAX_COUNT]	角点区敏感指数	[0,255]
graininess_sens[XMEDIA_ISP_ISO_MAX_COUNT]	颗粒性敏感指数	[0,255]
v_denoise_str[XMEDIA_ISP_ISO_MAX_COUNT]	控制值域核去噪强度	[0,255]
s_denoise_str[XMEDIA_ISP_ISO_MAX_COUNT]	控制空间核去噪强度。 XM7206V11A 不支持。	[0,255]
min_denoise_str[XMEDIA_ISP_ISO_MAX_COUNT]	最小去噪强度。 XM7606V13 不支持。	[0,511]
sigma[XMEDIA_ISP_ISO_MAX_COUNT]	滤波参数 sigma。 XM7606V13 不支持。	[0,127]
s_luma_thd_low[XMEDIA_ISP_ISO_MAX_COUNT]	根据运动信息调整空间核的去噪强度。	[0,255]
s_luma_thd_high[XMEDIA_ISP_ISO_MAX_COUNT]	如果运动强度小于 s_mot_thd_low 判定静止区，则用 s_mot_str_low 调整；	[0,255]
s_luma_str_low[XMEDIA_ISP_ISO_MAX_COUNT]	如果运动强度大于 s_mot_thd_high 判定运动区，则用 s_mot_str_high 调整；	[0,255]
s_luma_str_high[XMEDIA_ISP_ISO_MAX_COUNT]	如果运动强度介于两个阈值之间，则采用线性插值的方法得到调整量。 XM7206V11A 不支持。	[0,255]
s_mot_thd_low[XMEDIA_ISP_ISO_MAX_COUNT]	根据亮度调整空间核的去噪强度。	[0,255]
s_mot_thd_high[XMEDIA_ISP_ISO_MAX_COUNT]	如果图像亮度小于 s_luma_thd_low，则用 s_luma_str_low 调整；	[0,255]
s_mot_str_low[XMEDIA_ISP_ISO_MAX_COUNT]	如果图像亮度大于 s_luma_thd_high，则用 s_luma_str_high 调整；	[0,255]
s_mot_str_high[XMEDIA_ISP_ISO_MAX_COUNT]	如果图像亮度介于两个阈值之间，则采用线性插值的方法得到调整量。 XM7206V11A 不支持。	[0,255]
v_luma_thd_low[XMEDIA_ISP_ISO_MAX_COUNT]	根据亮度调整值域核的去噪强度。	[0,255]
v_luma_thd_high[XMEDIA_ISP_ISO_MAX_COUNT]	如果图像亮度小于 v_luma_thd_low，	[0,255]

成员名称	描述	取值范围
P_ISO_MAX_COUNT]	则用 v_luma_str_low 调整;	
v_luma_str_low[XMEDIA_ISP_ISO_MAX_COUNT]	如果图像亮度大于 v_luma_thd_high, 则用 v_luma_str_high 调整;	[0,255]
v_luma_str_high[XMEDIA_ISP_ISO_MAX_COUNT]	如果图像亮度介于两个阈值之间, 则采用线性插值的方法得到调整量。	[0,255]
v_mot_thd_low[XMEDIA_ISP_ISO_MAX_COUNT]	根据运动信息调整值域核的去噪强度。	[0,255]
v_mot_thd_high[XMEDIA_ISP_ISO_MAX_COUNT]	如果运动强度小于 v_mot_thd_low 判定静止区, 则用 v_mot_str_low 调整;	[0,255]
v_mot_str_low[XMEDIA_ISP_ISO_MAX_COUNT]	如果运动强度大于 v_mot_thd_high 判定运动区, 则用 v_mot_str_high 调整;	[0,255]
v_mot_str_high[XMEDIA_ISP_ISO_MAX_COUNT]	如果运动强度介于两个阈值之间, 则采用线性插值的方法得到调整量。	[0,255]
text_thd[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_3DNR_THD_MAX_COUNT]	根据纹理信息调整去噪强度。	[0,255]
text_str[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_3DNR_STR_MAX_COUNT]	如果小于 text_thd[0], 判定为平坦区, 用 text_str[0]强度; 如果大于 text_thd[1]且小于 text_thd[2], 判定为弱纹理区, 用 text_str[1]强度; 如果大于 text_thd[3], 判定为强边区, 用 text_str[2]强度。 如果介于 text_thd[0]和 text_thd[1]两个阈值之间, 则采用线性插值的方法得到调整量。 如果介于 text_thd[2]和 text_thd[3]两个阈值之间, 则采用线性插值的方法得到调整量。	[0,255]
luma_varscale_thd_low[XMEDIA_ISP_ISO_MAX_COUNT]	根据亮度调整方差大小。如果图像亮度小于 luma_varscale_thd_low, 则用 luma_varscale_str_low 调整; 如果图像亮度大于 luma_varscale_thd_high,	[0,255]
luma_varscale_thd_high[XMEDIA_ISP_ISO_MAX_COUNT]		[0,255]

成员名称	描述	取值范围
luma_varscale_str_low[XMEDIA_ISP_ISO_MAX_COUNT]	则用 luma_varscale_str_high 调整；如果图像亮度介于两个阈值之间，则采用线性插值的方法得到调整量。 XM7606V13 不支持。	[0,256]
luma_varscale_str_high[XMEDIA_ISP_ISO_MAX_COUNT]		[0,256]
var_scale[XMEDIA_ISP_ISO_MAX_COUNT]	对方差结果进行收缩。 XM7606V13 不支持。	[0,1023]
blend_var_thd_low[XMEDIA_ISP_ISO_MAX_COUNT]	根据方差大小调整两种滤波方式融合比例。如果方差小于 blend_var_thd_low，则用 YNR_blend_var_low_wgt 调整；如果方差大于 blend_var_thd_high，则用 blend_var_high_wgt 调整；如果方差介于两个阈值之间，则采用线性插值的方法得到调整量。 XM7606V13 不支持。	[0,255]
blend_var_thd_high[XMEDIA_ISP_ISO_MAX_COUNT]		[0,255]
blend_var_low_wgt[XMEDIA_ISP_ISO_MAX_COUNT]		[0,256]
blend_var_high_wgt[XMEDIA_ISP_ISO_MAX_COUNT]		[0,256]
blend_luma_thd_low[XMEDIA_ISP_ISO_MAX_COUNT]	根据亮度大小调整两种滤波方式融合比例。如果亮度小于 blend_luma_thd_low，则用 YNR_blend_luma_low_wgt 调整；如果亮度大于 blend_luma_thd_high，则用 blend_luma_high_wgt 调整；如果亮度介于两个阈值之间，则采用线性插值的方法得到调整量。 XM7606V13 不支持。	[0,255]
blend_luma_thd_high[XMEDIA_ISP_ISO_MAX_COUNT]		[0,255]
blend_luma_low_wgt[XMEDIA_ISP_ISO_MAX_COUNT]		[0,256]
blend_luma_high_wgt[XMEDIA_ISP_ISO_MAX_COUNT]		[0,256]
blend_mot_thd_low[XMEDIA_ISP_ISO_MAX_COUNT]	根据运动大小调整两种滤波方式融合比例。如果运动小于 blend_mot_thd_low，则用 blend_mot_low_wgt 调整；如果运动大于 blend_mot_thd_high，则用 blend_mot_high_wgt 调整；如果运动介于两个阈值之间，则采用线性插值的方法得到调整量。 XM7606V13 不支持。	[0,255]
blend_mot_thd_high[XMEDIA_ISP_ISO_MAX_COUNT]		[0,255]
blend_mot_low_wgt[XMEDIA_ISP_ISO_MAX_COUNT]		[0,256]
blend_mot_high_wgt[XMEDIA_ISP_ISO_MAX_COUNT]		[0,256]

成员名称	描述	取值范围
nr_addback_str[XMEDIA_ISP_ISO_MAX_COUNT]	噪声回叠强度。 XM7606V13 不支持。	[0,256]
blend_nr_center_var_thd_low[XMEDIA_ISP_ISO_MAX_COUNT]	在噪声回叠开启情况下，根据方差大小调整中心亮度和高斯滤波的融合比例。	[0,255]
blend_nr_center_var_thd_high[XMEDIA_ISP_ISO_MAX_COUNT]	如果方差小于 blend_nr_center_var_thd_low，则用 blend_nr_center_var_low_wgt 调整；	[0,255]
blend_nr_center_var_low_wgt[XMEDIA_ISP_ISO_MAX_COUNT]	如果方差大于 blend_nr_center_var_thd_high，则用 blend_nr_center_var_high_wgt 调整；	[0,256]
blend_nr_center_var_high_wgt[XMEDIA_ISP_ISO_MAX_COUNT]	如果方差介于两个阈值之间，则采用线性插值的方法得到调整量。 XM7606V13 不支持。	[0,256]
addback_text_thd[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_3DNR_THD_MAX_COUNT]	在噪声回叠开启情况下，根据纹理信息调整回叠强度。如果小于 addback_text_thd[0]，用	[0,255]
addback_text_str[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_3DNR_STR_MAX_COUNT]	addback_text_str[0]强度；如果大于 addback_text_thd[1]且小于 addback_text_thd[2]，用 addback_text_str[1]强度；如果大于 addback_text_thd[3]，用 addback_text_str[2]强度。 如果介于 addback_text_thd[0]和 addback_text_thd[1]两个阈值之间，则 采用线性插值的方法得到调整量。 如果介于 addback_text_thd[2]和 addback_text_thd[3]两个阈值之间，则 采用线性插值的方法得到调整量。 XM7606V13 不支持。	[0,511]
addback_luma_thd[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_3DNR_THD_MAX_COUNT]	在噪声回叠开启情况下，根据亮度调整回叠强度。如果小于 addback_luma_thd[0]，用	[0,255]
addback_luma_str[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_3DNR_STR_MAX_COUNT]	addback_luma_str[0]强度；如果大于 addback_luma_thd[1]且小于	[0,511]

成员名称	描述	取值范围
COUNT]	addback_luma_thd[2], 用 addback_luma_str[1]强度; 如果大于 addback_luma_thd[3], 用 addback_luma_str[2]强度。 如果介于 addback_luma_thd[0]和 addback_luma_thd[1]两个阈值之间, 则采用线性插值的方法得到调整量。 如果介于 addback_luma_thd[2]和 addback_luma_thd[3]两个阈值之间, 则采用线性插值的方法得到调整量。 XM7606V13 不支持。	
addback_text_luma_blend_mot_thd_low[XMEDIA_ISP_ISO_MAX_COUNT]	在噪声回叠开启情况下, 根据运动信息融合用纹理信息调整前和调整后回叠强度。 XM7606V13 不支持。	[0,255]
addback_text_luma_blend_mot_thd_high[XMEDIA_ISP_ISO_MAX_COUNT]		[0,256]
addback_mot_thd[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_3DNR_THD_MAX_COUNT]	在噪声回叠开启情况下, 根据运动信息调整回叠强度。如果小于 addback_mot_thd[0], 用 256 强度; 如果大于 addback_mot_thd[1]且小于 addback_mot_thd[2], 用 addback_mot_weak_str 强度; 如果大于 addback_mot_thd[3], 用 addback_mot_strong_str 强度。 如果介于 addback_mot_thd[0]和 addback_mot_thd[1]两个阈值之间, 则采用线性插值的方法得到调整量。 如果介于 addback_mot_thd[2]和 addback_mot_thd[3]两个阈值之间, 则采用线性插值的方法得到调整量。 XM7606V13 不支持。	[0,255]
addback_mot_weak_str[XMEDIA_ISP_ISO_MAX_COUNT]		[0,511]
addback_mot_strong_str[XMEDIA_ISP_ISO_MAX_COUNT]		[0,256]

【注意事项】

无。

【相关数据类型及接口】

[xmedia_isp_set_stnr_attr](#)

[xmedia_isp_get_stnr_attr](#)

xmedia_isp_stnr_tnr_param

【说明】

定义 tnr 属性。

【定义】

```
typedef struct {  
    xmedia_video_operation_mode    op_type;  
    xmedia\_isp\_stnr\_tnr\_param\_manual manual_attr;  
    xmedia\_isp\_stnr\_tnr\_param\_auto   auto_attr;  
} xmedia_isp_stnr_tnr_param;
```

【成员】

成员名称	描述
op_type;	tnr 操作模式, 0: 自动模式 1: 手动模式
manual_attr	tnr 手动参数结构
auto_attr	tnr 自动参数结构

【注意事项】

无。

【相关数据类型及接口】

[xmedia_isp_set_stnr_attr](#)

[xmedia_isp_get_stnr_attr](#)

xmedia_isp_stnr_tnr_param_manual

【说明】

定义 tnr 手动参数属性。

【定义】

typedef struct {

```
xmedia_u8  chroma_sens;
xmedia_u8  ghost_coring_uv;
xmedia_u8  ghost_coring;
xmedia_u8  md_win_size;
xmedia_u8  bgm_str;
xmedia_u8  str;
xmedia_u8  uv_str;
xmedia_u16 md_thd_low;
xmedia_u16 md_thd_high;
xmedia_u8  luma_md_thd[XMEDIA_ISP_3DNR_THD_MAX_COUNT];
xmedia_u16 luma_md_scl[XMEDIA_ISP_3DNR_STR_MAX_COUNT];
xmedia_u8  sat_md_thd_low;
xmedia_u8  sat_md_thd_high;
xmedia_u16 satl_md_scl;
xmedia_u16 sath_md_scl;
xmedia_u8  border_thd;
xmedia_u16 border_md_scl;
} xmedia_isp_stnr_tnr_param_manual;
```

【成员】

成员名称	描述	取值范围
chroma_sens	对运动区域判断的色度加权	[0,31]
ghost_coring_uv	UV 通道融合上限的 coring 值。 XM7206V11A 不支持。	[0,255]
ghost_coring	Y 通道融合上限的 coring 值。 XM7206V11A 不支持。	[0,255]
md_win_size	小图检测运动时所用的窗口大小	3,5,7,9
bgm_str	如果 tnr_bgm_mode 设置为 1(7206 直接生效), bgm_str 用于设置小图时域降噪强度, 值越大, 小图时域降噪强度越强	[0,255]
str	如果 tnr_bgm_mode 和 tnr_mode 设置为 1(7206 直接生效), str 用于设置 Y 通道时域降噪强度, 值越大, Y 通道时	[0,255]

成员名称	描述	取值范围
	域降噪强度越强	
uv_str	如果 tnr_bgm_mode 和 tnr_mode 设置为 1(7206 直接生效), uv_str 用于设置 UV 通道时域降噪强度, 值越大, UV 通道时域降噪强度越强	[0,255]
md_thd_low	当 tnr_bgm_mode 为 1 时(7206 无此限制), 这两个参数将生效。 运动检测结果小于 md_thd_low, 则认为是静止区域; 如果运动检测结果大于 md_thd_high, 则认为是运动区域。	[0,2047]
md_thd_high		[0,2047]
luma_md_thd[XMEDIA_ISP_3DNR_THD_MAX_COUNT] (XM7606V13 只用到 luma_md_thd[0]和 luma_md_thd[1])	根据亮度调整小图运动检测的结果。 对于 XM7606V13, 如果图像亮度小于 luma_md_thd[0], 则用 luma_md_scl[0]调整; 如果图像亮度大于 luma_md_thd[1], 则用 luma_md_scl[1]调整; 如果图像亮度介于 luma_md_thd[0]和 luma_md_thd[1]两个阈值之间, 则采用线性插值的方法得到调整量。 本版本 luma_md_thd[0], luma_md_thd[1], luma_md_scl[0], luma_md_scl[1]有效。 对于 XM7206V11A, 如果图像亮度小于 luma_md_thd[0], 则用 luma_md_scl[0]调整; 如果图像亮度大于 luma_md_thd[1], 小于 luma_md_thd[2],则用 luma_md_scl[1]调整; 如果图像亮度大于 luma_md_thd[3],	[0,255]
luma_md_scl[XMEDIA_ISP_3DNR_STR_MAX_COUNT] (XM7606V13 只用到 luma_md_scl [0]和 luma_md_scl [1])		[0,511]

成员名称	描述	取值范围
	则用 luma_md_scl[2]调整; 如果图像亮度介于 luma_md_thd[0]和 luma_md_thd[1]两个阈值之间, 则采用线性插值的方法得到调整量。 如果图像亮度介于 luma_md_thd[2]和 luma_md_thd[3]两个阈值之间, 则采用线性插值的方法得到调整量。	
sat_md_thd_low	根据饱和度调整小图运动检测的结果。	[0,255]
sat_md_thd_high	如果图像饱和度小于	[0,255]
satl_md_scl	sat_md_thd_low, 则用 satl_md_scl 调	[0,511]
sath_md_scl	整; 如果图像亮度大于 sat_md_thd_high, 则用 sath_md_scl 调整; 如果图像亮度介于两个阈值之间, 则采用线性插值的方法得到调整量。	[0,511]
border_thd	小图边界处理阈值。 XM7606V13 不支持。	[0,255]
border_md_scl	小图边界处理强度, 值越小越强。 XM7606V13 不支持。	[0,511]

【注意事项】

无。

【相关数据类型及接口】

[xmedia_isp_set_stnr_attr](#)

[xmedia_isp_get_stnr_attr](#)

xmedia_isp_stnr_tnr_param_auto

【说明】

定义 tnr 自动属性。

【定义】

```
typedef struct {
```

```

xmedia_u32 iso_num;
xmedia_u32 iso_level[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8 chroma_sens[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8 ghost_coring_uv[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8 ghost_coring[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8 md_win_size[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8 bgm_str[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8 str[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8 uv_str[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 md_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 md_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8
luma_md_thd[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_3DNR_THD_MAX_COUNT];
xmedia_u16
luma_md_scl[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_3DNR_STR_MAX_COUNT];
xmedia_u8 sat_md_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8 sat_md_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 satl_md_scl[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 sath_md_scl[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u8 border_thd[XMEDIA_ISP_ISO_MAX_COUNT];
xmedia_u16 border_md_scl[XMEDIA_ISP_ISO_MAX_COUNT];
} xmedia_isp_stnr_tnr_param_auto;

```

【成员】

成员名称	描述	取值范围
iso_num	ISO 等级	
iso_level[XMEDIA_ISP_ISO_MAX_COUNT]	ISO 数量	
chroma_sens[XMEDIA_ISP_ISO_MAX_COUNT]	对运动区域判断的色度加权	[0,31]
ghost_coring_uv[XMEDIA_ISP_ISO_MAX_COUNT]	UV 通道融合上限的 coring 值。 XM7206V11A 不支持。	[0,255]
ghost_coring[XMEDIA_ISP_ISO_MAX_COUNT]	Y 通道融合上限的 coring 值。 XM7206V11A 不支持。	[0,255]
md_win_size[XMEDIA_ISP_ISO_MAX_COUNT]	小图检测运动时所用的窗口大小	3,5,7,9
bgm_str[XMEDIA_ISP_ISO_MAX_COUNT]	如果 tnr_bgm_mode 设置为 1(7206 直接生效), bgm_str 用于设置小图时域	[0,255]

成员名称	描述	取值范围
	降噪强度，值越大，小图时域降噪强度越强	
str[XMEDIA_ISP_ISO_MAX_COUNT]	如果 tnr_bgm_mode 和 tnr_mode 设置为 1(7206 直接生效)，str 用于设置 Y 通道时域降噪强度，值越大，Y 通道时域降噪强度越强	[0,255]
uv_str[XMEDIA_ISP_ISO_MAX_COUNT]	如果 tnr_bgm_mode 和 tnr_mode 设置为 1(7206 直接生效)，uv_str 用于设置 UV 通道时域降噪强度，值越大，UV 通道时域降噪强度越强	[0,255]
md_thd_low[XMEDIA_ISP_ISO_MAX_COUNT]	当 tnr_bgm_mode 为 1 时(7206 无此限制)，这两个参数将生效。 运动检测结果小于 md_thd_low，则认为是静止区域； 如果运动检测结果大于 md_thd_high，则认为是运动区域。	[0,2047]
md_thd_high[XMEDIA_ISP_ISO_MAX_COUNT]		[0,2047]
luma_md_thd[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_3DNR_THD_MAX_COUNT] (XM7606V13 只用到 luma_md_thd[XMEDIA_ISP_ISO_MAX_COUNT][0]和 luma_md_thd[XMEDIA_ISP_ISO_MAX_COUNT][1])	对于 XM7606V13 ， 如果图像亮度小于 luma_md_thd[0]，则用 luma_md_scl[0]调整； 如果图像亮度大于 luma_md_thd[1]，则用 luma_md_scl[1]调整； 如果图像亮度介于 luma_md_thd[0]和 luma_md_thd[1]两个阈值之间，则采用线性插值的方法得到调整量。 本版本 luma_md_thd[0]，luma_md_thd[1]，luma_md_scl[0]，luma_md_scl[1]有效。 对于 XM7206V11A ， 如果图像亮度小于 luma_md_thd[0]，则用 luma_md_scl[0]调整；	[0,255]
luma_md_scl[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_3DNR_STR_MAX_COUNT] (XM7606V13 只用到 luma_md_scl[XMEDIA_ISP_ISO_MAX_COUNT][0]和 luma_md_scl[XMEDIA_ISP_ISO_MAX_COUNT][1])		[0,511]

成员名称	描述	取值范围
	如果图像亮度大于 luma_md_thd[1], 小于 luma_md_thd[2], 则用 luma_md_scl[1] 调整; 如果图像亮度大于 luma_md_thd[3], 则用 luma_md_scl[2] 调整; 如果图像亮度介于 luma_md_thd[0] 和 luma_md_thd[1] 两个阈值之间, 则采用线性插值的方法得到调整量。 如果图像亮度介于 luma_md_thd[2] 和 luma_md_thd[3] 两个阈值之间, 则采用线性插值的方法得到调整量。	
sat_md_thd_low[XMEDIA_ISP_ISO_MAX_COUNT]	根据饱和度调整小图运动检测的结果。	[0,255]
sat_md_thd_high[XMEDIA_ISP_ISO_MAX_COUNT]	如果图像饱和度小于 sat_md_thd_low, 则用 satl_md_scl 调整;	[0,255]
satl_md_scl[XMEDIA_ISP_ISO_MAX_COUNT]	如果图像亮度大于 sat_md_thd_high, 则用 sath_md_scl 调整;	[0,511]
sath_md_scl[XMEDIA_ISP_ISO_MAX_COUNT]	如果图像亮度介于两个阈值之间, 则采用线性插值的方法得到调整量。	[0,511]
border_thd[XMEDIA_ISP_ISO_MAX_COUNT]	小图边界处理阈值。XM7606V13 不支持。	[0,255]
border_md_scl[XMEDIA_ISP_ISO_MAX_COUNT]	小图边界处理强度, 值越小越强。XM7606V13 不支持。	[0,511]

【注意事项】

无

【相关数据类型及接口】

[xmedia_isp_set_stnr_attr](#)

[xmedia_isp_get_stnr_attr](#)

[xmedia_isp_stnr_cnr_param](#)

【说明】

定义 cnr 属性。

【定义】

```
typedef struct {  
    xmedia_video_operation_mode    op_type;  
    xmedia_isp_stnr_cnr_param_manual manual_attr;  
    xmedia_isp_stnr_cnr_param_auto  auto_attr;  
} xmedia_isp_stnr_cnr_param;
```

【成员】

成员名称	描述
op_type	cnr 操作模式, 0: 自动模式; 1: 手动模式
manual_attr	cnr 手动参数结构
auto_attr	cnr 自动参数结构

【注意事项】

无。

【相关数据类型及接口】

[xmedia_isp_set_stnr_attr](#)

[xmedia_isp_get_stnr_attr](#)

[xmedia_isp_stnr_cnr_param_manual](#)

【说明】

定义 cnr 手动参数属性。

【定义】

```
typedef struct {  
    xmedia_u8  struct_sens;  
    xmedia_u8  edge_det_thd;  
    xmedia_u8  edge_det_robust;  
    xmedia_u8  de_dark_color_str;  
    xmedia_u8  de_bg_color_spot_str;
```

```

xmedia_u8 coarse_denoise_str;
xmedia_u8 fine_denoise_str;
xmedia_u8 mot_str;
xmedia_u8 mot_denoise_thd_high;
xmedia_u8 mot_denoise_thd_low;
xmedia_u8 mot_denoise_str_high;
xmedia_u8 mot_denoise_str_low;
xmedia_u8 mot_uvnr_str;
xmedia_u16 mot_bld_str;
xmedia_u8 mot_bld_thd_low;
xmedia_u8 mot_bld_thd_high;
xmedia_u8 de_purple_edge_thd;
xmedia_u8 de_purple_edge_str;
xmedia_u8 purple_range;
xmedia_u8 spec_color_degree;
xmedia_u8 spec_color_str;
xmedia_u8 luma_thd_high;
xmedia_u8 luma_thd_low;
xmedia_u8 luma_str_high;
xmedia_u8 luma_str_low;
} xmedia_isp_stnr_cnr_param_manual;

```

【成员】

成员名称	描述	取值范围
struct_sens	滤波结构性敏感指数。	[0x0,0xFF]
edge_det_thd	边缘检测阈值。	[0x0,0x3F]
edge_det_robust	边缘检测抗噪阈值。	[0x0,0xFF]
de_dark_color_str	去暗部色噪强度。	[0x0,0xFF]
de_bg_color_spot_str	去背景色斑强度。	[0x0,0xFF]
coarse_denoise_str	去色噪强度（粗调）。	[0x0,0x3F]
fine_denoise_str	去色噪强度（细调）。	[0x0,0x3F]
mot_str	运动区域滤波强度。 XM7206v11A 版本不支持。	[0x0,0xFF]
mot_denoise_thd_high	运动区域降色噪高阈值。	[0x0,0xFF]

成员名称	描述	取值范围
mot_denoise_thd_low	运动区域降色噪低阈值。	[0x0,0xFF]
mot_denoise_str_high	运动区域降色噪强度。	[0x0,0xFF]
mot_denoise_str_low	静止区域降色噪强度。	[0x0,0xFF]
mot_uvnr_str	运动区域降色噪强度。	[0x0,0xFF]
mot_bld_str	运动区域融合强度。	[0x0,0x100]
mot_bld_thd_low	运动区域融合低阈值。	[0x0,0xFF]
mot_bld_thd_high	运动区域融合高阈值。	[0x0,0xFF]
de_purple_edge_thd	去紫边阈值。	[0x0,0x1F]
de_purple_edge_str	去紫边强度。	[0x0,0xFF]
purple_range	紫边范围。	[0x0,0xFF]
spec_color_degree	特殊色去噪角度。	[0x0,0xFF]
spec_color_str	特殊色去噪强度。	[0x0,0xFF]
luma_thd_high	亮度控制去噪高阈值。	[0x0,0xFF]
luma_thd_low	亮度控制去噪低阈值。	[0x0,0xFF]
luma_str_high	亮度控制去噪高强度。	[0x0,0xFF]
luma_str_low	亮度控制去噪低强度。	[0x0,0xFF]

【注意事项】

无。

【相关数据类型及接口】

[xmedia_isp_set_stnr_attr](#)

[xmedia_isp_get_stnr_attr](#)

xmedia_isp_stnr_cnr_param_auto

【说明】

定义 cnr 自动参数属性。

【定义】

```
typedef struct {
    xmedia_u32 iso_num;
    xmedia_u32 iso_level[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  struct_sens[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  edge_det_thd[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  edge_det_robust[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  de_dark_color_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  de_bg_color_spot_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  coarse_denoise_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  fine_denoise_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  mot_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  mot_denoise_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  mot_denoise_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  mot_denoise_str_high[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  mot_denoise_str_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  mot_uvnr_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16 mot_bld_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  mot_bld_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  mot_bld_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  de_purple_edge_thd[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  de_purple_edge_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  purple_range[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  spec_color_degree[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  spec_color_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  luma_thd_high[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  luma_thd_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  luma_str_high[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  luma_str_low[XMEDIA_ISP_ISO_MAX_COUNT];
} xmedia_isp_stnr_cnr_param_auto;
```

【成员】

成员名称	描述	取值范围
iso_num	ISO 个数。	[1, 16]
iso_level[XMEDIA_ISP_ISO_MAX_COUNT]	ISO 设定值。	[0x0,0xFFFFFFFF]
struct_sens[XMEDIA_ISP_ISO_MAX_COUNT]	滤波结构性敏感指数。	[0x0,0xFF]
edge_det_thd[XMEDIA_ISP_ISO_MAX_COUNT]	边缘检测阈值。	[0x0,0x3F]

成员名称	描述	取值范围
edge_det_robust[XMEDIA_ISP_ISO_MAX_COUNT]	边缘检测抗噪阈值。	[0x0,0xFF]
de_dark_color_str[XMEDIA_ISP_ISO_MAX_COUNT]	去暗部色噪强度。	[0x0,0xFF]
de_bg_color_spot_str[XMEDIA_ISP_ISO_MAX_COUNT]	去背景色斑强度。	[0x0,0xFF]
coarse_denoise_str[XMEDIA_ISP_ISO_MAX_COUNT]	去色噪强度（粗调）。	[0x0,0x3F]
fine_denoise_str[XMEDIA_ISP_ISO_MAX_COUNT]	去色噪强度（细调）。	[0x0,0x3F]
mot_str[XMEDIA_ISP_ISO_MAX_COUNT]	运动区域滤波强度。 XM7206v11A 版本不支持。	[0x0,0xFF]
mot_denoise_thd_high[XMEDIA_ISP_ISO_MAX_COUNT]	运动区域降色噪高阈值。	[0x0,0xFF]
mot_denoise_thd_low[XMEDIA_ISP_ISO_MAX_COUNT]	运动区域降色噪低阈值。	[0x0,0xFF]
mot_denoise_str_high[XMEDIA_ISP_ISO_MAX_COUNT]	运动区域降色噪强度。	[0x0,0xFF]
mot_denoise_str_low[XMEDIA_ISP_ISO_MAX_COUNT]	静止区域降色噪强度。	[0x0,0xFF]
mot_uvnr_str[XMEDIA_ISP_ISO_MAX_COUNT]	运动区域降色噪强度。	[0x0,0xFF]
mot_bld_str[XMEDIA_ISP_ISO_MAX_COUNT]	运动区域融合强度。	[0x0,0x100]
mot_bld_thd_low[XMEDIA_ISP_ISO_MAX_COUNT]	运动区域融合低阈值。	[0x0,0xFF]
mot_bld_thd_high[XMEDIA_ISP_ISO_MAX_COUNT]	运动区域融合高阈值。	[0x0,0xFF]
de_purple_edge_thd[XMEDIA_ISP_ISO_MAX_COUNT]	去紫边阈值。	[0x0,0x1F]
de_purple_edge_str[XMEDIA_ISP_ISO_MAX_COUNT]	去紫边强度。	[0x0,0xFF]
purple_range[XMEDIA_ISP_ISO_MAX_COUNT]	紫边范围。	[0x0,0xFF]
spec_color_degree[XMEDIA_ISP_ISO_MAX_COUNT]	特殊色去噪角度。	[0x0,0xFF]

成员名称	描述	取值范围
spec_color_str[XMEDIA_ISP_ISO_MAX_COUNT]	特殊色去噪强度。	[0x0,0xFF]
luma_thd_high[XMEDIA_ISP_ISO_MAX_COUNT]	亮度控制去噪高阈值。	[0x0,0xFF]
luma_thd_low[XMEDIA_ISP_ISO_MAX_COUNT]	亮度控制去噪低阈值。	[0x0,0xFF]
luma_str_high[XMEDIA_ISP_ISO_MAX_COUNT]	亮度控制去噪高强度。	[0x0,0xFF]
luma_str_low[XMEDIA_ISP_ISO_MAX_COUNT]	亮度控制去噪低强度。	[0x0,0xFF]

【注意事项】

无。

【相关数据类型及接口】

[xmedia_isp_set_stnr_attr](#)

[xmedia_isp_get_stnr_attr](#)

8.25 Sharpen

8.25.1 功能描述

Sharpen 模块用于增强图像的清晰度，可以实现对图像的带方向的边缘和无方向的细节纹理的单独锐化增强，而且，通过调节所要增强的频段，可以实现多种清晰度风格的增强效果。此外，还能控制锐化后的图像的 overshoot（白边白点）和 undershoot（黑边黑点）。

8.25.2 API 参考

- [xmedia_isp_set_sharpen_attr](#): 设定 Sharpen 属性。
- [xmedia_isp_get_sharpen_attr](#): 获取 Sharpen 属性。

[xmedia_isp_set_sharpen_attr](#)

【描述】

设定 Sharpen 属性。

【语法】

```
xmedia_s32 xmedia_isp_set_sharpen_attr(xmedia_u32 pipe, const xmedia_isp_sharpen_attr
*sharpen_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
sharpen_attr	sharpen_attr 属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

XM7606V13 此功能由 vpss 接口实现，参数同本节描述。

【举例】

无。

【相关主题】

[xmedia_isp_get_sharpen_attr](#)

xmedia_isp_get_sharpen_attr

【描述】

获取 Sharpen 属性。

【语法】

```
xmedia_s32 xmedia_isp_get_sharpen_attr(xmedia_u32 pipe, xmedia\_isp\_sharpen\_attr
*sharpen_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
sharpen_attr	sharpen_attr 属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: `xmedia_isp.h`
- 库文件: `libxmedia_isp.a`

【注意】

XM7606V13 此功能由 vpss 接口实现，参数同本节描述。

【举例】

无。

【相关主题】

[xmedia_isp_get_stnr_attr](#)

8.25.3 数据类型

- [xmedia_isp_sharpen_attr](#): 定义 Sharpen 模块属性。
- [xmedia_isp_sharpen_ysb_combine_mode](#): 定义 sharpen 细节融合模式属性。
- [xmedia_isp_sharpen_manual_attr](#): 定义 sharpen 手动属性。
- [xmedia_isp_sharpen_auto_attr](#): 定义 sharpen 自动属性。

xmedia_isp_sharpen_attr

【说明】

定义 sharpen 属性。

【定义】

```
typedef struct {  
    xmedia_bool                enable;  
    xmedia_isp_sharpen_ysh_combine_mode detail_combine_mode;  
    xmedia_s8                  skin_ub[XMEDIA_ISP_SHARPEN_SKIN_NUM];  
    xmedia_s8                  skin_lb[XMEDIA_ISP_SHARPEN_SKIN_NUM];  
    xmedia_u8                  skin_delta[XMEDIA_ISP_SHARPEN_SKIN_NUM];  
    xmedia_u8                  rgb_delta[XMEDIA_ISP_SHARPEN_RGB_NUM];  
    xmedia_video_operation_mode op_type;  
    xmedia_isp_sharpen_manual_attr manual_attr;  
    xmedia_isp_sharpen_auto_attr  auto_attr;  
} xmedia_isp_sharpen_attr;
```



说明

宏定义如下：

```
#define XMEDIA_ISP_SHARPEN_SKIN_NUM    2  
#define XMEDIA_ISP_SHARPEN_RGB_NUM    3
```

【成员】

成员名称	描述
enable	Sharpen 模块开关，取值范围[0, 1]。
detail_combine_mode	Sharpen 细节融合模式，取值范围[0, 1]。
skin_ub[XMEDIA_ISP_SHARPEN_SKIN_NUM]	肤色区域矩形窗 UV 上边界，取值范围[-64, 63]。
skin_lb[XMEDIA_ISP_SHARPEN_SKIN_NUM]	肤色区域矩形窗 UV 下边界，取值范围[-64, 63]。
skin_delta[XMEDIA_ISP_SHARPEN_SKIN_NUM];	肤色区域矩形窗收缩步长，取值范围[0, 7]。
rgb_delta[XMEDIA_ISP_SHARPEN_RGB_NUM]	RGB 颜色区域收缩步长，取值范围[0, 31]。
op_type	Sharpen 工作模式。
manual_attr	Sharpen 手动模式属性。
auto_attr	Sharpen 自动模式属性。

【注意事项】

无。

【相关数据类型及接口】

[xmedia_isp_set_sharpen_attr](#)

[xmedia_isp_get_sharpen_attr](#)

xmedia_isp_sharpen_ysh_combine_mode

【说明】

定义 Sharpen 细节合成模式相关属性。

【定义】

```
typedef enum {
```

```

XMEDIA_ISP_SHARPEN_YSH_COMBINE_MODE_FUSION = 0,
XMEDIA_ISP_SHARPEN_YSH_COMBINE_MODE_FUSION_WITH_WEIGHT,
XMEDIA_ISP_SHARPEN_YSH_COMBINE_MODE_MAX
} xmedia_isp_sharpen_ysh_combine_mode;

```

【成员】

成员名称	描述
XMEDIA_ISP_SHARPEN_YSH_COMBINE_MODE_FUSION	Fusion 合成模式，此时最终细节取 edge 与 texture 直接叠加结果。
XMEDIA_ISP_SHARPEN_YSH_COMBINE_MODE_FUSION_WITH_WEIGHT	权重融合模式，此时最终细节取 edge 与 texture 加权融合结果。

【注意事项】

无。

【相关数据类型及接口】

[xmedia_isp_set_sharpen_attr](#)

[xmedia_isp_get_sharpen_attr](#)

[xmedia_isp_sharpen_manual_attr](#)

【说明】

定义 sharpen 手动参数属性。

【定义】

```

typedef struct {
    xmedia_u8  edge_luma_wgt[XMEDIA_ISP_SHARPEN_LUMA_NUM];
    xmedia_u8  tex_luma_wgt[XMEDIA_ISP_SHARPEN_LUMA_NUM];
    xmedia_u8  edge_freq;
    xmedia_u8  tex_freq;
    xmedia_u16 edge_str;
    xmedia_u16 tex_str;
    xmedia_s8  pos_edge_lut[XMEDIA_ISP_SHARPEN_GAIN_NUM];
    xmedia_s8  neg_edge_lut[XMEDIA_ISP_SHARPEN_GAIN_NUM];
    xmedia_s8  pos_tex_lut[XMEDIA_ISP_SHARPEN_GAIN_NUM];
    xmedia_s8  neg_tex_lut[XMEDIA_ISP_SHARPEN_GAIN_NUM];
    xmedia_u8  edge_dering_str[XMEDIA_ISP_SHARPEN_DERING_NUM];
}

```

```

xmedia_u8 tex_dering_str[XMEDIA_ISP_SHARPEN_DERING_NUM];
xmedia_u8 dir_dering_str[XMEDIA_ISP_SHARPEN_DERING_NUM];
xmedia_u8 max_sharp_gain;
xmedia_u8 edge_coring;
xmedia_u8 rgain;
xmedia_u8 ggain;
xmedia_u8 bgain;
xmedia_u8 skin_gain;
xmedia_u8 global_overshoot_str;
xmedia_u8 global_undershoot_str;
xmedia_u8 global_overshoot_thd;
xmedia_u8 global_undershoot_thd;
xmedia_u16 region_division_sensitivity;
xmedia_u16 region_division_thd[XMEDIA_ISP_SHARPEN_VAR_THD_NUM];
xmedia_u8 region_division_str[XMEDIA_ISP_SHARPEN_VAR_STR_NUM];
xmedia_u8 dir_edge_mode;
xmedia_u8 dir_sensitivity;
xmedia_u8 dir_edge_freq;
xmedia_u8 dir_edge_thd[XMEDIA_ISP_SHARPEN_DIR_THD_NUM];
xmedia_u8 dir_edge_str[XMEDIA_ISP_SHARPEN_DIR_STR_NUM];
xmedia_u8 fusion_edge_max_edge;
xmedia_u8 fusion_edge_max_flat;
xmedia_u8 fusion_edge_thd;
xmedia_u8 fusion_flat_thd;
} xmedia_isp_sharpen_manual_attr;

```



说明

宏定义如下：

```

#define XMEDIA_ISP_SHARPEN_LUMA_NUM      33
#define XMEDIA_ISP_SHARPEN_GAIN_NUM      16
#define XMEDIA_ISP_SHARPEN_DERING_NUM    17
#define XMEDIA_ISP_SHARPEN_VAR_THD_NUM   4
#define XMEDIA_ISP_SHARPEN_DIR_THD_NUM   4
#define XMEDIA_ISP_SHARPEN_VAR_STR_NUM    3
#define XMEDIA_ISP_SHARPEN_DIR_STR_NUM    3

```

【成员】

成员名称	描述
edge_luma_wgt[XMEDIA_ISP_SHARPEN_LUMA_NUM]	edge 锐化强度随亮度的联动曲线，取值范围[0, 16]。 XM7606V13 不支持。

成员名称	描述
tex_luma_wgt[XMEDIA_ISP_SHARPEN_LUMA_NUM]	texture 锐化强度随亮度的联动曲线, 取值范围[0, 16]。 XM7606V13 不支持。
edge_freq	edge 锐化频段选择, 取值范围[0, 1]。
tex_freq	texture 锐化频段选择, 取值范围[0, 1]。
edge_str	edge 全局锐化强度, 取值范围[0, 255]。
tex_str	texture 全局锐化强度, 取值范围[0, 255]。
pos_edge_lut[XMEDIA_ISP_SHARPEN_GAIN_NUM]	正向 edge 根据本身细节强度的非线性锐化强度, 取值范围[-31, 31]。
neg_edge_lut[XMEDIA_ISP_SHARPEN_GAIN_NUM]	负向 edge 根据本身细节强度的非线性锐化强度, 取值范围[-31, 31]。
pos_tex_lut[XMEDIA_ISP_SHARPEN_GAIN_NUM]	正向 texture 根据本身细节强度的非线性锐化强度, 取值范围[-31, 31]。
neg_tex_lut[XMEDIA_ISP_SHARPEN_GAIN_NUM]	负向 texture 根据本身细节强度的非线性锐化强度, 取值范围[-31, 31]。
edge_dering_str[XMEDIA_ISP_SHARPEN_DERING_NUM]	edge 锐化后 shoot 控制强度随细节强度变化曲线, 取值范围[0, 64]。
tex_dering_str[XMEDIA_ISP_SHARPEN_DERING_NUM]	texture 锐化后 shoot 控制强度随细节强度变化曲线, 取值范围[0, 64]。
dir_dering_str[XMEDIA_ISP_SHARPEN_DERING_NUM]	方向性细节锐化后 shoot 控制强度随细节强度变化曲线, 取值范围[0, 64]。 XM7606V13 不支持。
max_sharp_gain	最大锐化增益, 取值范围[0, 255]。 XM7606V13 不支持。
edge_coring	edge 整体 coring 强度, 取值范围[0, 255]。
rgain	R 颜色通道单独锐化强度, 取值范围[0, 31]。

成员名称	描述
ggain	G 颜色通道单独锐化强度, 取值范围[0, 31]。
bgain	B 颜色通道单独锐化强度, 取值范围[0, 31]。
skin_gain	肤色区域单独锐化强度, 取值范围[0, 31]。
global_overshoot_str	全局白边控制强度, 取值范围[0, 255]。
global_undershoot_str	全局黑边控制强度, 取值范围[0, 255]。
global_overshoot_thd	白边控制阈值, 锐化后数值高于阈值部分做白边抑制处理, 取值范围[0, 255]。
global_undershoot_thd	黑边控制阈值, 锐化后数值低于阈值部分做黑边抑制处理, 取值范围[0, 255]。
region_division_sensitivity	区域划分检测强度, 取值范围[0, 511]。 XM7606V13 不支持。
region_division_thd[XMEDIA_ISP_SHARPEN_VAR_THD_NUM]	区域划分阈值, 取值范围[0, 511]。 XM7606V13 不支持。
region_division_str[XMEDIA_ISP_SHARPEN_VAR_STR_NUM]	区域锐化强度曲线, 取值范围[0, 255]。 XM7606V13 不支持。
dir_edge_mode	方向性细节回叠模式选择, 取值范围[0, 2]。 XM7606V13 不支持。
dir_sensitivity	方向性细节检测强度, 取值范围[0, 255]。 XM7606V13 不支持。
dir_edge_freq	方向性细节频段选择, 取值范围[0, 1]。 XM7606V13 不支持。
dir_edge_thd[XMEDIA_ISP_SHARPEN_DIR_THD_NUM]	方向性细节阈值参数, 取值范围[0, 255]。 XM7606V13 不支持。
dir_edge_str[XMEDIA_ISP_SHARPEN_DIR_STR_NUM]	方向性细节锐化强度选择,

成员名称	描述
	取值范围[0, 255]。 XM7606V13 不支持。
fusion_edge_max_edge	边缘区细节的最大回叠权重， 取值范围[0, 255]。
fusion_edge_max_flat	平坦区细节的最大回叠权重， 取值范围[0, 255]。
fusion_edge_thd	边缘区阈值，取值范围[0, 255]。
fusion_flat_thd	平坦区阈值，取值范围[0, 255]。

【注意事项】

无。

【相关数据类型及接口】

[xmedia_isp_set_sharpen_attr](#)

[xmedia_isp_get_sharpen_attr](#)

[xmedia_isp_sharpen_auto_attr](#)

【说明】

定义 sharpen 自动参数属性。

【定义】

```
typedef struct {
    xmedia_u8    iso_num;
    xmedia_u32 iso_level[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8
edge_luma_wgt[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_LUMA_NUM];
    xmedia_u8
tex_luma_wgt[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_LUMA_NUM];
    xmedia_u8    edge_freq[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8    tex_freq[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16 edge_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16 tex_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_s8
pos_edge_lut[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_GAIN_NUM];
    xmedia_s8
```

```

neg_edge_lut[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_GAIN_NUM];
    xmedia_s8
pos_tex_lut[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_GAIN_NUM];
    xmedia_s8
neg_tex_lut[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_GAIN_NUM];
    xmedia_u8
edge_dering_str[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_DERING_NUM];
    xmedia_u8
tex_dering_str[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_DERING_NUM];
    xmedia_u8
dir_dering_str[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_DERING_NUM];
    xmedia_u8  max_sharp_gain[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  edge_coring[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  rgain[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  ggain[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  bgain[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  skin_gain[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  global_overshoot_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  global_undershoot_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  global_overshoot_thd[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  global_undershoot_thd[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16 region_division_sensitivity[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u16
region_division_thd[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_VAR_THD_NUM
];
    xmedia_u8
region_division_str[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_VAR_STR_NUM]
;
    xmedia_u8 dir_sensitivity[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8 dir_edge_freq[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8 dir_edge_thd[XMEDIA_ISP_ISO_MAX_COUNT]
                [XMEDIA_ISP_SHARPEN_DIR_THD_NUM];
    xmedia_u8
dir_edge_str[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_DIR_STR_NUM];
    xmedia_u8 fusion_edge_max_edge[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8 fusion_edge_max_flat[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8 fusion_edge_thd[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8 fusion_flat_thd[XMEDIA_ISP_ISO_MAX_COUNT];
} xmedia_isp_sharpen_auto_attr;

```

【成员】

成员名称	描述
iso_num	ISO 数量。
iso_level[XMEDIA_ISP_ISO_MAX_COUNT]	ISO 等级。
edge_luma_wgt[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_LUMA_NUM]	edge 锐化强度随亮度的联动曲线，取值范围[0, 16]。 XM7606V13 不支持。
tex_luma_wgt[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_LUMA_NUM]	texture 锐化强度随亮度的联动曲线，取值范围[0, 16]。 XM7606V13 不支持。
edge_freq[XMEDIA_ISP_ISO_MAX_COUNT]	edge 锐化频段选择，取值范围[0, 1]。
tex_freq[XMEDIA_ISP_ISO_MAX_COUNT]	texture 锐化频段选择，取值范围[0, 1]。
edge_str[XMEDIA_ISP_ISO_MAX_COUNT]	edge 全局锐化强度，取值范围[0, 255]。
tex_str[XMEDIA_ISP_ISO_MAX_COUNT]	texture 全局锐化强度，取值范围[0, 255]。
pos_edge_lut[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_GAIN_NUM]	正向 edge 根据本身细节强度的非线性锐化强度，取值范围[-31, 31]。
neg_edge_lut[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_GAIN_NUM]	负向 edge 根据本身细节强度的非线性锐化强度，取值范围[-31, 31]。
pos_tex_lut[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_GAIN_NUM]	正向 texture 根据本身细节强度的非线性锐化强度，取值范围[-31, 31]。
neg_tex_lut[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_GAIN_NUM]	负向 texture 根据本身细节强度的非线性锐化强度，取值范围[-31, 31]。
edge_dering_str[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_DERING_NUM]	edge 锐化后 shoot 控制强度随细节强度变化曲线，取值范围[0, 64]。
tex_dering_str[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_DERING_NUM]	texture 锐化后 shoot 控制强度随细节强度变化曲线，取值范围[0, 64]。
dir_dering_str[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_DERING_NUM]	方向性细节锐化后 shoot 控制强度随细节强度变化曲线，取值范围[0, 64]。

成员名称	描述
	XM7606V13 不支持。
max_sharp_gain[XMEDIA_ISP_ISO_MAX_COUNT]	最大锐化增益, 取值范围[0, 255]。 XM7606V13 不支持。
edge_coring[XMEDIA_ISP_ISO_MAX_COUNT]	edge 整体 coring 强度, 取值范围[0, 255]。
rgain[XMEDIA_ISP_ISO_MAX_COUNT]	R 颜色通道单独锐化强度, 取值范围[0, 31]。
ggain[XMEDIA_ISP_ISO_MAX_COUNT]	G 颜色通道单独锐化强度, 取值范围[0, 31]。
bgain[XMEDIA_ISP_ISO_MAX_COUNT]	B 颜色通道单独锐化强度, 取值范围[0, 31]。
skin_gain[XMEDIA_ISP_ISO_MAX_COUNT]	肤色区域单独锐化强度, 取值范围[0, 31]。
global_overshoot_str[XMEDIA_ISP_ISO_MAX_COUNT]	全局白边控制强度, 取值范围[0, 255]。
global_undershoot_str[XMEDIA_ISP_ISO_MAX_COUNT]	全局黑边控制强度, 取值范围[0, 255]。
global_overshoot_thd[XMEDIA_ISP_ISO_MAX_COUNT]	白边控制阈值, 锐化后数值高于阈值部分做白边抑制处理, 取值范围[0, 255]。
global_undershoot_thd[XMEDIA_ISP_ISO_MAX_COUNT]	黑边控制阈值, 锐化后数值低于阈值部分做黑边抑制处理, 取值范围[0, 255]。
region_division_sensitivity[XMEDIA_ISP_ISO_MAX_COUNT]	区域划分检测强度, 取值范围[0, 511]。 XM7606V13 不支持。
region_division_thd[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_VAR_THD_NUM]	区域划分阈值, 取值范围[0, 511]。 XM7606V13 不支持。
region_division_str[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_VAR_THD_NUM]	区域锐化强度曲线, 取值范围[0, 255]。

成员名称	描述
SHARPEN_VAR_STR_NUM]	XM7606V13 不支持。
dir_sensitivity[XMEDIA_ISP_ISO_MAX_COUNT]	方向性细节检测强度，取值范围[0, 255]。 XM7606V13 不支持。
dir_edge_freq[XMEDIA_ISP_ISO_MAX_COUNT]	方向性细节频段选择，取值范围[0, 1]。 XM7606V13 不支持。
dir_edge_thd[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_DIR_THD_NUM]	方向性细节阈值参数，取值范围[0, 255]。 XM7606V13 不支持。
dir_edge_str[XMEDIA_ISP_ISO_MAX_COUNT][XMEDIA_ISP_SHARPEN_DIR_STR_NUM]	方向性细节锐化强度选择，取值范围[0, 255]。 XM7606V13 不支持。
fusion_edge_max_edge[XMEDIA_ISP_ISO_MAX_COUNT]	边缘区细节的最大回叠权重， 取值范围[0, 255]。
fusion_edge_max_flat[XMEDIA_ISP_ISO_MAX_COUNT]	平坦区细节的最大回叠权重， 取值范围[0, 255]。
fusion_edge_thd[XMEDIA_ISP_ISO_MAX_COUNT]	边缘区阈值，取值范围[0, 255]。
fusion_flat_thd[XMEDIA_ISP_ISO_MAX_COUNT]	平坦区阈值，取值范围[0, 255]。

【注意事项】

无。

【相关数据类型及接口】

[xmedia_isp_set_sharpen_attr](#)

[xmedia_isp_get_sharpen_attr](#)

8.26 AFC

8.26.1 功能描述

AFC 为去伪彩模块，该模块支持图像高频区域的伪彩进行去除。

8.26.2 API 参考

- `xmedia_isp_set_anti_false_color_attr`: 设定 anti false color 属性。
- `xmedia_isp_get_anti_false_color_attr`: 获取 anti false color 属性。

`xmedia_isp_set_anti_false_color_attr`

【描述】

设定 AFC 属性。

【语法】

```
xmedia_s32 xmedia_isp_set_anti_false_color_attr(xmedia_u32
pipe, const xmedia_isp_anti_false_color_attr *anti_false_color_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
anti_false_color_attr	anti_false_color_attr 属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: `xmedia_isp.h`
- 库文件: `libxmedia_isp.a`

【注意】

XM7606V13 暂未支持。

【举例】

无。

【相关主题】

[xmedia_isp_get_anti_false_color_attr](#)

xmedia_isp_get_anti_false_color_attr

【描述】

获取 AFC 属性。

【语法】

```
xmedia_s32 xmedia_isp_get_anti_false_color_attr(xmedia_u32 pipe,  
xmedia\_isp\_anti\_false\_color\_attr *anti_false_color_attr);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
anti_false_color_attr	anti_false_color_attr 属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

XM7606V13 暂未支持。

【举例】

无。

【相关主题】

[xmedia_isp_set_anti_false_color_attr](#)

8.26.3 数据类型

- [xmedia_isp_anti_false_color_attr](#): 定义 AFC 模块属性。
- [xmedia_isp_anti_false_color_manual_attr](#): 定义 AFC 手动属性。
- [xmedia_isp_anti_false_color_auto_attr](#): 定义 AFC 自动属性。

[xmedia_isp_anti_false_color_attr](#)

【说明】

定义 AFC 模块属性。

【定义】

```
typedef struct {  
    xmedia_bool enable;  
    xmedia_video_operation_mode op_type;  
    xmedia\_isp\_anti\_false\_color\_auto\_attr auto_attr;  
    xmedia\_isp\_anti\_false\_color\_manual\_attr manual_attr;  
} xmedia_isp_anti_false_color_attr;
```

【成员】

成员名称	描述
enable	AFC 模块使能开关, 0: 关闭; 1: 使能。
op_type	操作类型。 0: 自动; 1: 手动。
auto_attr	AFC 自动参数属性。
manual_attr	AFC 手动参数属性。

【注意事项】

XM7606V13 暂未支持。

【相关数据类型及接口】

[xmedia_isp_set_anti_false_color_attr](#)

[xmedia_isp_get_anti_false_color_attr](#)

xmedia_isp_anti_false_color_manual_attr

【说明】

定义 AFC 手动参数。

【定义】

```
typedef struct {  
    xmedia_u8 edge_thd;  
    xmedia_u8 edge_gain;  
    xmedia_u8 fcr_str;  
    xmedia_u8 sat_low;  
    xmedia_u8 sat_high;  
} xmedia_isp_anti_false_color_manual_attr;
```

【成员】

成员名称	描述	取值范围
edge_thd	去伪彩阈值。	[0x0,0x1F]
edge_gain	去伪彩边缘增益。	[0x0,0xFF]
fcr_str	去伪彩强度。	[0x0,0xFF]
sat_low	去伪彩饱和度低阈值。	[0x0,0xFF]
sat_high	去伪彩饱和度高阈值。	[0x0,0xFF]

【注意事项】

XM7606V13 暂未支持。

【相关数据类型及接口】

[xmedia_isp_set_anti_false_color_attr](#)

[xmedia_isp_get_anti_false_color_attr](#)

xmedia_isp_anti_false_color_auto_attr

【说明】

定义 AFC 自动参数属性。

【定义】

```
typedef struct {
    xmedia_u8  iso_num;
    xmedia_u32 iso_level[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  edge_thd[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  edge_gain[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  fcr_str[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  sat_low[XMEDIA_ISP_ISO_MAX_COUNT];
    xmedia_u8  sat_high[XMEDIA_ISP_ISO_MAX_COUNT];
} xmedia_isp_anti_false_color_attr;
```

【成员】

成员名称	描述	取值范围
iso_num	ISO 个数。	[0x1, 0x10]
iso_level[XMEDIA_ISP_ISO_MAX_COUNT]	ISO 设定值。	[0x0,0xFFFFFFFF]
edge_thd[XMEDIA_ISP_ISO_MAX_COUNT]	不同 ISO 下的去伪彩阈值。	[0x0,0x1F]
edge_gain[XMEDIA_ISP_ISO_MAX_COUNT]	不同 ISO 下的去伪彩边缘增益。	[0x0,0xFF]
fcr_str[XMEDIA_ISP_ISO_MAX_COUNT]	不同 ISO 下的去伪彩饱和度低阈值。	[0x0,0xFF]
sat_low[XMEDIA_ISP_ISO_MAX_COUNT]	不同 ISO 下的去伪彩饱和度低阈值。	[0x0,0xFF]
sat_high[XMEDIA_ISP_ISO_MAX_COUNT]	不同 ISO 下的去伪彩饱和度高阈值。	[0x0,0xFF]

【注意事项】

XM7606V13 暂未支持。

【相关数据类型及接口】

[xmedia_isp_set_sharpen_attr](#)

[xmedia_isp_get_sharpen_attr](#)

9 统计信息

9.1 概述

ISP 提供的 3A 统计信息及相关配置。

9.2 功能描述

9.2.1 AE 统计信息

实现自动曝光信息的统计，软件根据统计信息调节 sensor 可实现自动曝光的功能。
AE 统计信息包含分区间的 R/Gr/B/Gb 均值统计、带权重全局 R/Gr/B/Gb 均值、1024 段直方图信息。

9.2.1.1 AE 区域统计信息

AE 最大支持 24×16 分块，最小支持 1×1 分块，每个分块均可以输出 16bit R, Gr, B, Gb 均值(Average R/Gr/B/Gb)。

9.2.1.2 带权重的 AE 全局统计信息

与 AE 的区域统计信息基本一致，对于整幅图像，AE 也提供了 4 个全局统计信息。分别为 `global_avg[0]`, `global_avg[1]`, `global_avg[2]`, `global_avg[3]`，分别代表通道 R,Gr,B,Gb。其物理意义与分块统计一致。

9.2.1.3 带权重的 AE 直方图信息

AE 提供 1024 阶直方图统计 0-1023，直方图信息读取地址与数据的对应关系如表 9-1 所示。

表9-1 AE 直方图统计信息

Memory address	Zone	Memory Read Data	
		[31:29]	[28:0]
0	0	0	Pixel Value 为 0 的个数
1	1	0	Pixel Value 为 1 的个数
2	2	0	Pixel Value 为 2 的个数
3	3	0	Pixel Value 为 3 的个数
.....			
1022	1022	0	Pixel Value 为 1022 的个数
1023	1023	0	Pixel Value 为 1023 的个数

AE 支持 hist_four_mode 模式选择，在 hist_four_mode = 0 或 hist_four_mode = 1 模式下，AE 统计逐像素或单一通道的 1024bin，如上图所示，在 hist_four_mode = 2 时，AE 提供四通道 256 段直方图统计 0~255，直方图信息读取地址与数据的对应关系如表 9-2 所示。

表9-2 AE 直方图统计信息

Memory address	Zone	Memory Read Data	
		[31:29]	[28:0]
0	0	0	R 通道 Pixel Value 为 0 的个数
1	1	0	R 通道 Pixel Value 为 1 的个数
.....			
255	255	0	R 通道 Pixel Value 为 255 的个数
256	256	0	Gr 通道 Pixel Value 为 0 的个数
.....			
511	511	0	Gr 通道 Pixel Value 为 255 的个数

Memory address	Zone	Memory Read Data	
		[31:29]	[28:0]
512	512	0	B 通道 Pixel Value 为 0 的个数
.....			
767	767	0	B 通道 Pixel Value 为 255 的个数
768	768	0	Gb 通道 Pixel Value 为 0 的个数
.....			
1022	1022	0	Gb 通道 Pixel Value 为 254 的个数
1023	1023	0	Gb 通道 Pixel Value 为 255 的个数

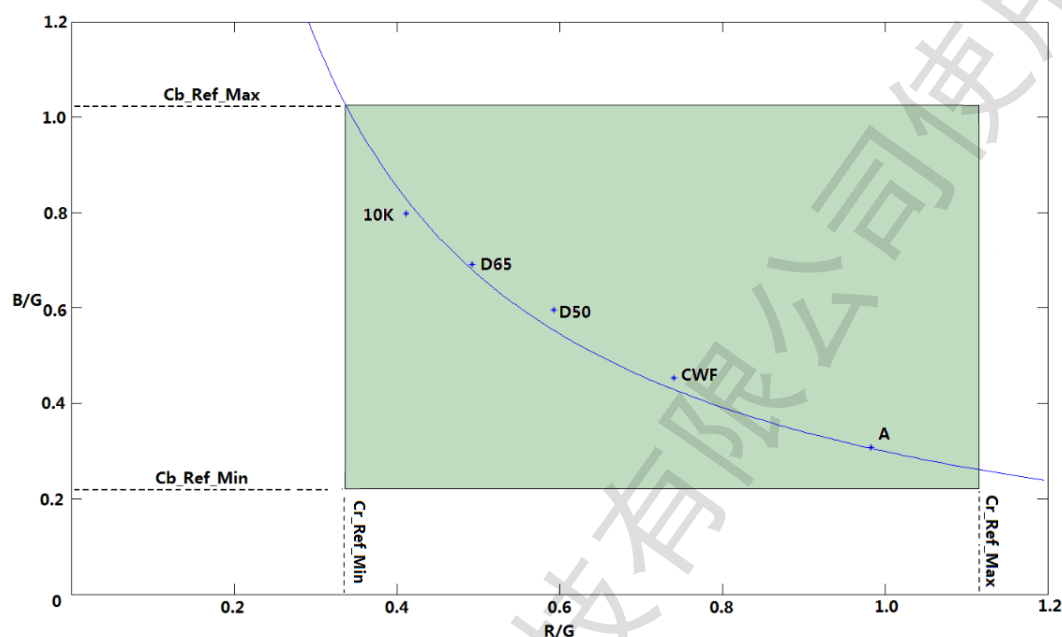
9.2.2 AWB 统计信息

AWB 统计信息包含全局统计信息和区域统计信息。

- 全局统计信息：整幅图像的 R, G, B 均值, 以及有效统计点的个数。
- 区域统计信息：最大支持图像的 32x32 分块, 每个分块均可以输出 R, G, B 均值, 以及有效统计点的个数。

AWB 中统计的有效点, 除了要满足配置的 RGB 上限与下限要求外 (black_level, white_level), 像素点颜色还需要满足配置的有效像素颜色空间, 具体如图 9-1 所示, 绿色区块内的为有效像素的颜色范围。

图9-1 AWB 有效像素颜色空间



AWB 统计的有效像素区域是一个四边形，其中上下左右四个边界由 `cr_ref_min/max`, `cb_ref_min/max` 这 4 个参数决定，它们代表了允许的 `r/g`, `b/g` 的最大与最小值。

9.2.2.1 AWB 区域统计信息

AWB 中最大支持图像的 32x32 分块，每个分块均可以输出 R、G、B 均值 (`zone_avg_r/zone_avg_g/zone_avg_b`)。其中统计的像素点均为 16bit，分块内统计有效点个数 `zone_valid_count`。

须知

- 当分辨率较小，分块数较多时，如果每块的分辨率小于 60x14，那么需要减少分块数，以免每块过小。建议在分辨率小的情况下，使用较少的分块数。
- 如果要输出低分辨率，需要在业务启动的时候设置一个比较小的 AWB 统计信息分块数。

9.2.2.2 AWB 全局统计信息

与 AWB 的区域统计信息基本一致，对于整幅图像，AWB 也提供了四个全局统计信息。分别为 `global_r`, `global_g`, `global_b`, 以及有效点统计 `global_valid_count`。其物理意义与分块统计一致。

9.3 API 参考

统计信息的 API 接口必须在调用 xmedia_isp_init 接口之后才能调用。

- [xmedia_isp_set_statistics_config](#): 设置 ISP 3A 统计信息配置。
- [xmedia_isp_get_statistics_config](#): 获取 ISP 3A 统计信息配置。
- [xmedia_isp_get_ae_statistics](#): 获取 AE 统计信息。
- [xmedia_isp_get_ae_blend_statistics](#): 获取拼接模式 AE 统计信息。
- [xmedia_isp_get_awb_statistics](#): 获取 AWB 统计信息。
- [xmedia_isp_get_awb_blend_statistics](#): 获取拼接模式 AWB 统计信息。
- [xmedia_isp_get_af_statistics](#): 获取 AF 统计信息。

xmedia_isp_set_statistics_config

【描述】

设置 ISP 3A 统计信息配置。

【语法】

```
xmedia_s32 xmedia_isp_set_statistics_config(xmedia_u32 pipe, const xmedia\_isp\_statistics\_cfg
*stat_cfg);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
stat_cfg	ISP 统计信息配置。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

用户可以关闭不需要的统计信息，以提高系统性能。

【相关主题】

[xmedia_isp_get_statistics_config](#)

xmedia_isp_get_statistics_config

【描述】

获取 ISP 3A 统计信息配置。

【语法】

```
xmedia_s32 xmedia_isp_get_statistics_config(xmedia_u32 pipe, xmedia_isp_statistics_cfg
*stat_cfg);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
stat_cfg	ISP 统计信息配置。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【相关主题】

[xmedia_isp_set_statistics_config](#)

xmedia_isp_get_ae_statistics

【描述】

获取 AE 统计信息。

【语法】

```
xmedia_s32 xmedia_isp_get_ae_statistics(xmedia_u32 pipe, xmedia\_isp\_ae\_statistics *ae_stat);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ae_stat	AE 统计信息。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【相关主题】

无

xmedia_isp_get_ae_blend_statistics

【描述】

获取拼接模式 AE 统计信息。

【语法】

```
xmedia_s32 xmedia_isp_get_ae_blend_statistics(xmedia_u32 pipe,
xmedia_isp_ae_blend_statistics *ae_blend_stat);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
ae_blend_stat	拼接模式 AE 统计信息。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【相关主题】

无

xmedia_isp_get_awb_statistics

【描述】

获取 WB 统计信息。

【语法】

```
xmedia_s32 xmedia_isp_get_awb_statistics(xmedia_u32 pipe, xmedia_isp_awb_statistics
*awb_stat);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
awb_stat	WB 统计信息。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【相关主题】

无

xmedia_isp_get_awb_blend_statistics

【描述】

获取拼接模式下 WB 统计信息。

【语法】

```
xmedia_s32 xmedia_isp_get_awb_blend_statistics(xmedia_u32 pipe,
xmedia_isp_awb_blend_statistics *awb_blend_stat);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
awb_blend_stat	拼接模式 WB 统计信息。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【相关主题】

无

xmedia_isp_get_af_statistics

【描述】

获取 AF 统计信息。

【语法】

```
xmedia_s32 xmedia_isp_get_af_statistics(xmedia_u32 pipe, xmedia_isp_af_statistics *af_stat);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
af_stat	AF 统计信息。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_isp.h
- 库文件: libxmedia_isp.a

【注意】

无

【相关主题】

无

9.4 数据类型

- [xmedia_isp_statistics_cfg](#): 定义 ISP 3A 统计信息配置。
- [xmedia_isp_ae_switch](#): 定义 AE 统计模块在 ISP pipeline 的位置。
- [xmedia_isp_ae_hist_mode](#): 定义 AE 分区模式。
- [xmedia_isp_ae_hist_four_mode](#): 直方图统计模式选择。
- [xmedia_isp_ae_hist_offset](#): 直方图统计时水平和垂直方向起始点设置。
- [xmedia_isp_ae_hist_skip](#): 直方图统计时采样点设置。
- [xmedia_isp_ae_hist_config](#): 定义全局直方图统计时的采样点方式。
- [xmedia_isp_ae_single_color_bin](#): 定义颜色通道选择。

- `xmedia_isp_ae_stat_mode`: 定义统计模块的开方模式。
- `xmedia_isp_ae_crop`: 对 AE 模块输入图像进行裁剪。
- `xmedia_isp_ae_stat_cfg`: 定义 AE 统计信息配置。
- `xmedia_isp_ae_statistics_cfg`: 定义 AE 统计信息配置。
- `xmedia_isp_awb_switch`: 定义 AWB 统计模块在 ISP pipeline 的位置。
- `xmedia_isp_awb_crop`: 对 AWB 模块输入图像进行裁剪。
- `xmedia_isp_awb_statistics_cfg`: 定义 AWB 统计信息配置。
- `xmedia_isp_af_switch`: 定义 AE 统计模块在 ISP pipeline 的位置。
- `xmedia_isp_af_crop`: 对 AF 模块输入图像进行裁剪。
- `xmedia_isp_af_raw_cfg`: raw 域统计前进行 pre_gamma 的曲线配置。
- `xmedia_isp_af_cfg`: 定义 AF 统计信息配置。
- `xmedia_isp_af_filter_mode`: 统计时计算方式选择。
- `xmedia_isp_af_ld`: AF Level Depend Gain 模块配置。
- `xmedia_isp_af_coring`: AF Coring 模块配置。
- `xmedia_isp_af_h_param`: AF 统计水平滤波器 IIR 参数设置。
- `xmedia_isp_af_v_param`: AF 统计垂直滤波器 IIR 参数设置。
- `xmedia_isp_fv_param`: AF 统计结果输出格式参数。
- `xmedia_isp_af_statistics_cfg`: 定义 AF 统计信息配置。
- `xmedia_isp_ae_grid_info`: 定义 AE 分区间统计信息的坐标信息。
- `xmedia_isp_ae_statistics`: 定义 AE 统计信息。
- `xmedia_isp_ae_blend_statistics`: 定义拼接模式 AE 统计信息。
- `xmedia_isp_awb_grid_info`: 定义 AWB 分区间统计信息的坐标信息。
- `xmedia_isp_awb_statistics`: 定义 AWB 统计信息。
- `xmedia_isp_awb_blend_statistics`: 定义拼接模式 AWB 统计信息。
- `xmedia_isp_af_zone`: 定义 AF 分块统计输出。
- `xmedia_isp_af_grid_info`: 定义 AF 分区间统计信息的坐标信息。
- `xmedia_isp_fe_af_statistics`: 定义 ISP AF 的 FE 统计信息。
- `xmedia_isp_be_af_statistics`: 定义 ISP AF 的 BE 统计信息。
- `xmedia_isp_af_statistics`: 定义 ISP AF 统计信息。

xmedia_isp_statistics_cfg

【说明】

定义 ISP 3A 统计信息配置。

【定义】

```
typedef struct {  
    xmedia_isp_ae_statistics_cfg  ae_cfg;  
    xmedia_isp_awb_statistics_cfg awb_cfg;  
    xmedia_isp_af_statistics_cfg  af_cfg;  
} xmedia_isp_statistics_cfg;
```

【成员】

成员名称	描述
ae_cfg	AE 统计信息配置。
awb_cfg	AWB 统计信息配置。
af_cfg	AF 统计信息配置。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_ae_switch

【说明】

定义 AE 统计模块在 ISP pipeline 的位置。

【定义】

```
typedef enum {  
    XMEDIA_ISP_AE_SWITCH_AFTER_DGN2 = 0,  
    XMEDIA_ISP_AE_SWITCH_AFTER_DGN3,  
    XMEDIA_ISP_AE_SWITCH_AFTER_DRC,  
    XMEDIA_ISP_AE_SWITCH_AFTER_MAX,  
} xmedia_isp_ae_switch;
```

【成员】

成员名称	描述
XMEDIA_ISP_AE_SWITCH_AFTER_DGN2	After ISP digital gain。
XMEDIA_ISP_AE_SWITCH_AFTER_DGN3	After static WB。
XMEDIA_ISP_AE_SWITCH_AFTER_DRC	After DRC。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_ae_hist_mode

【说明】

定义 AE 分区模式。

【定义】

```
typedef enum {
    XMEDIA_ISP_AE_HIST_MODE_NO_WEIGHT = 1,
    XMEDIA_ISP_AE_HIST_MODE_WEIGHT,
    XMEDIA_ISP_AE_HIST_MODE_MAX,
} xmedia_isp_ae_hist_mode;
```

【成员】

成员名称	描述
XMEDIA_ISP_AE_HIST_MODE_NO_WEIGHT	直方图不带权重的全图统计。
XMEDIA_ISP_AE_HIST_MODE_WEIGHT	直方图只统计分块区内，且带分块权重。

【注意事项】

无。

【相关数据类型及接口】

无

xmedia_isp_ae_hist_offset

【说明】

直方图统计时水平和垂直方向起始点设置。

【定义】

```
typedef enum {  
    XMEDIA_ISP_AE_HIST_OFFSET_ONE = 0,  
    XMEDIA_ISP_AE_HIST_OFFSET_TWO,  
    XMEDIA_ISP_AE_HIST_OFFSET_MAX,  
} xmedia_isp_ae_hist_offset;
```

【成员】

成员名称	描述
XMEDIA_ISP_AE_HIST_OFFSET_ONE	从图像的第一行/第一列开始统计。
XMEDIA_ISP_AE_HIST_OFFSET_TWO	从图像的第二行/第二列开始统计。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_ae_hist_skip

【说明】

直方图统计时采样点设置。

【定义】

```
typedef enum {  
    XMEDIA_ISP_AE_HIST_SKIP_EVERY_PIXEL = 0,  
    XMEDIA_ISP_AE_HIST_SKIP_EVERY_2ND_PIXEL,  
    XMEDIA_ISP_AE_HIST_SKIP_EVERY_3RD_PIXEL,  
    XMEDIA_ISP_AE_HIST_SKIP_EVERY_4TH_PIXEL,  
    XMEDIA_ISP_AE_HIST_SKIP_EVERY_5TH_PIXEL,  
    XMEDIA_ISP_AE_HIST_SKIP_EVERY_8TH_PIXEL,  
    XMEDIA_ISP_AE_HIST_SKIP_EVERY_9TH_PIXEL,  
    XMEDIA_ISP_AE_HIST_SKIP_MAX,  
} xmedia_isp_ae_hist_skip;
```

【成员】

成员名称	描述
XMEDIA_ISP_AE_HIST_SKIP_EVERY_PIXEL	图像中每个点都统计。
XMEDIA_ISP_AE_HIST_SKIP_EVERY_2ND_PIXEL	图像在水平或垂直方向上每两个点统计一次。
XMEDIA_ISP_AE_HIST_SKIP_EVERY_3RD_PIXEL	图像在水平或垂直方向上每三个点统计一次。
XMEDIA_ISP_AE_HIST_SKIP_EVERY_4TH_PIXEL	图像在水平或垂直方向上每四个点统计一次。
XMEDIA_ISP_AE_HIST_SKIP_EVERY_5TH_PIXEL	图像在水平或垂直方向上每五个点统计一次。
XMEDIA_ISP_AE_HIST_SKIP_EVERY_8TH_PIXEL	图像在水平或垂直方向上每八个点统计一次。
XMEDIA_ISP_AE_HIST_SKIP_EVERY_9TH_PIXEL	图像在水平或垂直方向上每九个点统计一次。

【注意事项】

hist 采样只在 hist_fourmode=0 时才生效。

【相关数据类型及接口】

无

xmedia_isp_ae_hist_config

【说明】

定义全局直方图统计时的采样点方式。

【定义】

```
typedef struct {  
    xmedia_isp_ae_hist_skip    hist_skip_x;  
    xmedia_isp_ae_hist_skip    hist_skip_y;  
    xmedia_isp_ae_hist_offset hist_offset_x;  
    xmedia_isp_ae_hist_offset hist_offset_y;  
} xmedia_isp_ae_hist_config;
```

【成员】

成员名称	描述
hist_skip_x	直方图统计时水平方向采样点设置, 0=every pixel; 1=every 2nd pixel; 2=every 3rd pixel; 3=every 4th pixel; 4=every 5th pixel; 5=every 8th pixel; 6=every 9th pixel。即取值为 0 时表示每 1 个像素采样一个进行统计, 取值为 1 时表示每 2 个像素采样一个进行统计, 以此类推。
hist_skip_y	直方图统计时垂直方向采样点设置, 0=every pixel; 1=every 2nd pixel; 2=every 3rd pixel; 3=every 4th pixel; 4=every 5th pixel; 5=every 8th pixel; 6=every 9th pixel。
hist_offset_x	直方图统计时水平方向起始点设置, 0 表示从第一列开始统计, 1 表示从第二列开始统计。
hist_offset_y	直方图统计时垂直方向起始点设置, 0 表示从第一行开始统计, 1 表示从第二行开始统计。

【注意事项】

- 可根据 Sensor 输入的 Bayer 数据格式, 合理配置该结构体, 使得全局直方图统计时只计入某单个分量数据。
- RGBIR 模式推荐全局直方图统计时只计入 G 分量, 可根据 RGBIR Sensor 的 Bayer 数据格式合理配置该结构体实现。
- 配置 stHistConfig 相关参数时, 需要把 XMEDIA_AE 库的大面积单色判断功能关闭。
- hist 采样只在 hist_fourmode=0 时才生效。

【相关数据类型及接口】

无

xmedia_isp_ae_hist_four_mode

【说明】

直方图统计模式选择。可选逐像素统计、单一颜色 1024bin 统计、4 通道(rgbgb)256bin 组合成 1024bin 统计。

【定义】

```
typedef enum {  
    XMEDIA_ISP_AE_HIST_FOUR_MODE_EVERY_COLOR = 0,
```

```

XMEDIA_ISP_AE_HIST_FOUR_MODE_SINGLE_COLOR,
XMEDIA_ISP_AE_HIST_FOUR_MODE_FOUR_PLANE,
XMEDIA_ISP_AE_HIST_FOUR_MODE_MAX,
} xmedia_isp_ae_hist_four_mode;

```

【成员】

成员名称	描述
XMEDIA_ISP_AE_HIST_FOUR_MODE_EVERY_COLOR	所有颜色逐个统计。
XMEDIA_ISP_AE_HIST_FOUR_MODE_SINGLE_COLOR	只统计单一颜色。
XMEDIA_ISP_AE_HIST_FOUR_MODE_FOUR_PLANE	4 颜色通道单独统计。

【注意事项】

无。

【相关数据类型及接口】

无

xmedia_isp_ae_single_color_bin

【说明】

定义颜色通道选择。

【定义】

```

typedef enum {
    XMEDIA_ISP_AE_SINGLE_COLOR_BIN_R = 0,
    XMEDIA_ISP_AE_SINGLE_COLOR_BIN_GR,
    XMEDIA_ISP_AE_SINGLE_COLOR_BIN_B,
    XMEDIA_ISP_AE_SINGLE_COLOR_BIN_GB,
    XMEDIA_ISP_AE_SINGLE_COLOR_BIN_MAX,
} xmedia_isp_ae_single_color_bin;

```

【成员】

成员名称	描述
XMEDIA_ISP_AE_SINGLE_COLOR_BIN_R	单一颜色统计选择 R 通道。
XMEDIA_ISP_AE_SINGLE_COLOR_BIN_GR	单一颜色统计选择 GR 通道。
XMEDIA_ISP_AE_SINGLE_COLOR_BIN_B	单一颜色统计选择 B 通道。
XMEDIA_ISP_AE_SINGLE_COLOR_BIN_GB	单一颜色统计选择 GB 通道。

【注意事项】

无。

【相关数据类型及接口】

无

xmedia_isp_ae_stat_mode

【说明】

定义统计模块的开方模式。

【定义】

```
typedef enum {
    XMEDIA_ISP_AE_STAT_MODE_NORMAL = 0,
    XMEDIA_ISP_AE_STAT_MODE_ROOT,
    XMEDIA_ISP_AE_STAT_MODE_MAX,
} xmedia_isp_ae_stat_mode;
```

【成员】

成员名称	描述
XMEDIA_ISP_AE_STAT_MODE_NORMAL	普通模式，不作处理。
XMEDIA_ISP_AE_STAT_MODE_ROOT	开方模式。

【注意事项】

无。

【相关数据类型及接口】

无

xmedia_isp_ae_crop

【说明】

对 AE 模块输入图像进行裁剪。

【定义】

```
typedef struct {  
    xmedia_bool enable;  
    xmedia_u16  x;  
    xmedia_u16  y;  
    xmedia_u16  w;  
    xmedia_u16  h;  
} xmedia_isp_ae_crop;
```

【成员】

成员名称	描述
enable	AE CROP 模块的使能。 0: 不使能 Crop; 1: 使能 Crop;
x	Crop x 起始位置。 取值范围: [0, image_width - MAX_COL_NUM*2];
y	Crop y 起始位置。 取值范围: [0, image_height - MAX_ROW_NUM*2];
w	Crop 宽度。 取值范围: [MAX_COL_NUM*2, image_width];
h	Crop 高度。 取值范围: [MAX_ROW_NUM*2, image_height];

【注意事项】

- 水平方向起始位置与 Crop 宽度之和应小于图像宽度，垂直方向起始位置与 Crop 高度之和应小于图像高度。
- 对于 BE 的 AE 统计信息，ISP BE 分块场景下，不支持 AE CROP（输入图像裁剪）功能。

【相关数据类型及接口】

无

xmedia_isp_ae_stat_cfg

【说明】

定义 AE 统计信息配置。

【定义】

```
typedef struct {
    xmedia_bool          global_avg_en;
    xmedia_bool          zone_avg_en;
    xmedia_bool          hist_en;
    xmedia_isp_ae_switch pos_switch;
    xmedia_isp_ae_hist_mode hist_mode;
    xmedia_isp_ae_hist_four_mode hist_four_mode;
    xmedia_isp_ae_hist_config hist_config;
    xmedia_isp_ae_single_color_bin single_color_bin;
    xmedia_isp_ae_stat_mode hist_root_mode;
    xmedia_isp_ae_stat_mode aver_root_mode;
    xmedia_isp_ae_crop crop;
    xmedia_u16 h_zone_num;
    xmedia_u16 v_zone_num;
    xmedia_u8
weight[XMEDIA_ISP_AE_ZONE_ROW_MAX_NUM][XMEDIA_ISP_AE_ZONE_COL_MAX_NUM];
} xmedia_isp_ae_stat_cfg;
```

说明

宏定义如下：

```
#define XMEDIA_ISP_AE_ZONE_ROW_MAX_NUM    16
#define XMEDIA_ISP_AE_ZONE_COL_MAX_NUM    24
```

【成员】

成员名称	描述
global_avg_en	全局均值使能。
zone_avg_en	分块均值使能。
hist_en	直方图使能。
pos_switch	Be ae 位置选择。
hist_mode	直方图模式。
hist_four_mode	直方图统计模式选择。 0 : every color 1: single_color 1:256bin rggb 拼成 1024bin
hist_config	跳点采样。
single_color_bin	当 hist_four_mode=1 时, 选择哪个通道统计直方图 0-3: rbgg
hist_root_mode	均值开方模式。0 为不开方, 1 为开方。仅影响 BE 均值统计信息。
aver_root_mode	均值开方模式。 0; 关 1: 开。仅影响 BE 均值统计信息
crop	AE 输入图像裁剪配置。
h_zone_num	水平方向分块个数[1,16]。
v_zone_num	垂直方向分块个数[1,24]。
weight	16*24 个区间的 AE 权重表, 取值范围[0,15]。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_ae_statistics_cfg

【说明】

定义 AE 统计信息配置。

【定义】

```
typedef struct {  
    xmedia_isp_ae_stat_cfg fe_cfg;  
    xmedia_isp_ae_stat_cfg be_cfg;  
} xmedia_isp_ae_statistics_cfg;
```

【成员】

成员名称	描述
fe_cfg	AE 在 FE 中的统计信息配置。
be_cfg	AE 在 BE 中的统计信息配置。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_awb_switch

【说明】

定义 AWB 统计模块在 ISP pipeline 的位置。

【定义】

```
typedef enum {  
    XMEDIA_ISP_AWB_SWITCH_AFTER_EXPANDER = 0,  
    XMEDIA_ISP_AWB_SWITCH_AFTER_DG2,  
    XMEDIA_ISP_AWB_SWITCH_AFTER_DRC,  
    XMEDIA_ISP_AWB_SWITCH_AFTER_MAX,  
} xmedia_isp_awb_switch;
```

【成员】

成员名称	描述
XMEDIA_ISP_AWB_SWITCH_AFTER_EXPANDER	AWB 统计模块位置在 expander Gain 后。
XMEDIA_ISP_AWB_SWITCH_AFTER_DG2	AWB 统计模块位置在 DGN2 后。

XMEDIA_ISP_AWB_SWITCH_AFTER_DRC	AWB 统计模块位置在 DRC 后。
---------------------------------	--------------------

【注意事项】

无。

【相关数据类型及接口】

无

xmedia_isp_awb_crop

【说明】

对 AWB 模块输入图像进行裁剪。

【定义】

```
typedef struct {
    xmedia_bool enable;
    xmedia_u16  x;
    xmedia_u16  y;
    xmedia_u16  w;
    xmedia_u16  h;
} xmedia_isp_awb_crop;
```

【成员】

成员名称	描述
enable	使能 Crop, 0 为 Crop 不使能, 1 为 Crop 使能。
x	Crop X 起始位置。 取值范围: [0, image_width - MAX_COL_NUM*2]
y	Crop Y 起始位置。 取值范围: [0, image_height - MAX_ROW_NUM*2]
w	Crop 宽度。 取值范围: [MAX_COL_NUM*2, image_width]
h	Crop 高度。 取值范围: [MAX_ROW_NUM*2, image_height]

【注意事项】

- 水平方向起始位置与 Crop 宽度之和应小于图像宽度，垂直方向起始位置与 Crop 高度之和应小于图像高度。
- ISP BE 分块场景下，不支持 AWB CROP（输入图像裁剪）功能。

【相关数据类型及接口】

无

xmedia_isp_awb_statistics_cfg

【说明】

定义 AWB 统计信息配置。

【定义】

```
typedef struct {
    xmedia_bool      global_avg_en;
    xmedia_bool      zone_avg_en;
    xmedia_isp_awb_switch pos_switch;
    xmedia_u16        h_zone_num;
    xmedia_u16        v_zone_num;
    xmedia_u16        white_level;
    xmedia_u16        black_level;
    xmedia_u16        cb_max;
    xmedia_u16        cb_min;
    xmedia_u16        cr_max;
    xmedia_u16        cr_min;
    xmedia_isp_awb_crop crop;
    xmedia_u8         pseudo_wgt[XMEDIA_ISP_AWB_PSEUDO_WEIGHT_NUM];
} xmedia_isp_awb_statistics_cfg;
```



说明

宏定义如下：

```
#define XMEDIA_ISP_AWB_PSEUDO_WEIGHT_NUM    4
```

【成员】

成员名称	描述
global_avg_en	全局均值使能
zone_avg_en	分块均值使能

pos_switch	WB 统计模块位置，默认值为 0，在 Digital Gain 后。
h_zone_num	WB 统计水平方向分块个数，取值范围：[1,32], BlkNum 为 ISP BE 的分块数目。
v_zone_num	WB 统计垂直方向分块个数。取值范围：[1, 32]。
white_level	统计白点信息时，找白点的亮度上限。取值范围：[0x0, 0xFFFF]，默认值 0xFFFF。
black_level	统计白点信息时，找白点的亮度下限。取值范围：[0x0, u16WhiteLevel]，默认值 0x0。
cb_max	统计白点信息时，色差 B/G 的最大值，8bit 精度，默认值 512。取值范围：[0x0, 0xFFFF]。
cb_min	统计白点信息时，色差 B/G 的最小值，8bit 精度，默认值 128。取值范围：[0x0, 0xFFFF]。
cr_max	统计白点信息时，色差 R/G 的最大值，8bit 精度，默认值 512。取值范围：[0x0, 0xFFFF]。
cr_min	统计白点信息时，色差 R/G 的最小值，8bit 精度，默认值 128。取值范围：[0x0, 0xFFFF]。
crop	AWB 输入图像裁剪配置。
pseudo_wgt	假 y 计算权重，取值范围[0,15]

【注意事项】

无。

【相关数据类型及接口】

无

xmedia_isp_af_switch

【说明】

定义 AF 统计模块在 ISP pipeline 的位置。

【定义】

```
typedef enum {
```

```

XMEDIA_ISP_AF_AFTER_DGN2 = 0,
XMEDIA_ISP_AF_AFTER_DRC,
XMEDIA_ISP_AF_AFTER_CSC,
XMEDIA_ISP_AF_AFTER_MAX,
} xmedia_isp_af_switch;

```

【成员】

成员名称	描述
XMEDIA_ISP_AF_AFTER_DGN2	将 AF 模块放在 DGain 后进行统计，属于 RAW 域。
XMEDIA_ISP_AF_AFTER_DRC	将 AF 模块放在 DRC 后进行统计，属于 RAW 域。
XMEDIA_ISP_AF_AFTER_CSC	将 AF 模块放在 CSC 后进行统计，属于 YUV 域。

【注意事项】

- WDR 模式下，非宽动态场景的 bayer 域（即 RAW 域）统计信息无明显变化趋势，推荐使用 YUV 域统计信息。

【相关数据类型及接口】

无

xmedia_isp_af_crop

【说明】

对 AF 模块输入图像进行裁剪。

【定义】

```

typedef struct {
    xmedia_bool enable;
    xmedia_u16  x;
    xmedia_u16  y;
    xmedia_u16  w;
    xmedia_u16  h;
} xmedia_isp_af_crop;

```

【成员】

成员名称	描述
enable	使能 Crop, 取值范围: [0, 1]
x	crop X 起始位置, 取值范围: [0, image_width - MAX_COL_NUM*2]
y	crop Y 起始位置, 取值范围: [0, image_height - MAX_ROW_NUM*2]
w	crop 宽度, 取值范围: [MAX_COL_NUM*2, image_width]
h	crop 高度, 取值范围: [MAX_ROW_NUM*2, image_height]

【注意事项】

- 水平方向起始位置与 crop 宽度之和应小于图像宽度，垂直方向起始位置与 Crop 高度之和应小于图像高度。
- 对于 BE 的 AF 统计信息，ISP BE 分块场景下，不支持 AF CROP（输入图像裁剪）功能。

【相关数据类型及接口】

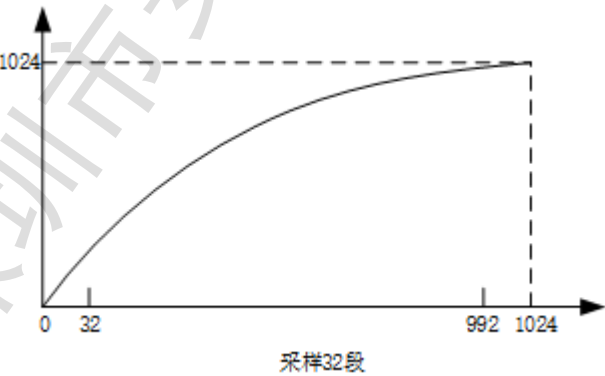
无

xmedia_isp_af_raw_cfg

【说明】

raw 域统计前进行 pre_gamma 的曲线配置。

图9-2 pre_gamma 的曲线示意图



【定义】

```
typedef struct {
```

```
xmedia_u16 tone_curve[XMEDIA_ISP_AF_TONE_CURVE_NUM];
} xmedia_isp_af_raw_cfg;
```

 说明
宏定义如下：

```
#define XMEDIA_ISP_AF_TONE_CURVE_NUM    32
```

【成员】

成员名称	描述
tone_curve	raw 域统计前进行 pre_gamma 的曲线配置。 当统计位置为 CSC 位置时，tone_curve 设置不生效。

【注意事项】

无。

【相关数据类型及接口】

无

xmedia_isp_af_cfg

【说明】

定义 AF 统计信息配置。

【定义】

```
typedef struct {
    xmedia_bool    pdaf_stat_en;
    xmedia_bool    cdaf_stat_en;
    xmedia_isp_af_switch pos_switch;
    xmedia_u16     h_zone_num;
    xmedia_u16     v_zone_num;
    xmedia_u16     h_size;
    xmedia_u16     v_size;
    xmedia_u16     hlt_thd;
    xmedia_u8      h_iir_wgt[XMEDIA_ISP_AF_FILTER_WEIGHT_NUM];
    xmedia_u8      v_fir_wgt[XMEDIA_ISP_AF_FILTER_WEIGHT_NUM];
    xmedia_s16     pre_filter_en;
    xmedia_isp_af_crop crop;
    xmedia_isp_af_raw_cfg raw_cfg;
} xmedia_isp_af_cfg;
```



宏定义如下：

```
#define XMEDIA_ISP_AF_FILTER_WEIGHT_NUM      2
```

【成员】

成员名称	描述
pdaf_stat_en	pdaf 统计使能。
cdaf_stat_en	cdaf 统计使能(普通 AF 统计模式)。
pos_switch	AF 统计信息位置配置，决定是在 Bayer 还是 YUV 域统计。
h_zone_num	AF 统计水平方向窗口个数，取值范围：[1, 18]。
v_zone_num	AF 统计垂直方向窗口个数，取值范围：[1, 15]。
h_size	AF 统计输入图像宽度。
v_size	AF 统计输入图像高度。
hlt_thd	高亮点最低亮度阈值。
h_iir_wgt	水平滤波器加权重。
v_fir_wgt	垂直滤波器加权重。
pre_filter_en	预滤波器模式， 0：关 1：开
crop	AF 输入图像的裁剪配置。
raw_cfg	亮度映射曲线，提升 raw 统计亮度辅助滤波器计算。

【注意事项】

无。

【相关数据类型及接口】

无

xmedia_isp_af_filter_mode

【说明】

统计时计算方式选择，可选逐点累加模式，或峰值模式(每行最大值累加)。

【定义】

```
typedef enum {  
    XMEDIA_ISP_AF_FILTER_MODE_NORM = 0,  
    XMEDIA_ISP_AF_FILTER_MODE_PEAK,  
    XMEDIA_ISP_AF_FILTER_MODE_MAX,  
} xmedia_isp_af_filter_mode;
```

【成员】

成员名称	描述
XMEDIA_ISP_AF_FILTER_MODE_NORM	滤波器模式：求和
XMEDIA_ISP_AF_FILTER_MODE_PEAK	滤波器模式：峰值

【注意事项】

无。

【相关数据类型及接口】

无

xmedia_isp_af_ld

【说明】

AF Level Depend Gain 模块配置。

【定义】

```
typedef struct {  
    xmedia_u8 gain_curve[XMEDIA_ISP_AF_GAIN_CURVE_NUM];  
} xmedia_isp_af_ld;
```

 说明

宏定义如下：

```
#define XMEDIA_ISP_AF_GAIN_CURVE_NUM    16
```

【成员】

成员名称	描述
gain_curve	Level depend gain，各个亮度下的 gain 强度，详解见 12.4 AF 统计信息使用说明。

【注意事项】

无。

【相关数据类型及接口】

无

xmedia_isp_af_coring

【说明】

AF Coring 模块配置。

【定义】

```
typedef struct {  
    xmedia_s16 coring_en;  
    xmedia_s16 coring_th;  
    xmedia_s16 coring_slp;  
    xmedia_s16 coring_lmt;  
} xmedia_isp_af_coring;
```

【成员】

成员名称	描述
coring_en	Coring 使能
coring_th	设置 Coring Threshold, 取值范围: [0x0, 0x7FF]。
coring_slp	设置 Coring Slope, 取值范围: [0x0, 0xF]。
coring_lmt	设置 Coring Maximum Output, 取值范围: [0x0, 0x7FF]。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_af_h_param

【说明】

AF 统计水平滤波器 IIR 参数设置。

【定义】

```
typedef struct {
    xmedia_bool          iir_en[XMEDIA_ISP_AF_IIR_EN_NUM];
    xmedia_s16           bias_offset;
    xmedia_u16           min_thd;
    xmedia_s16           iir_gain[XMEDIA_ISP_AF_IIR_GAIN_NUM];
    xmedia_s16           forward_coeff[XMEDIA_ISP_AF_IIR_FORWARD_COEF_NUM];
    xmedia_s16           backward_coeff[XMEDIA_ISP_AF_IIR_BACKWARD_COEF_NUM];
    xmedia_isp_af_filter_mode filter_mode;
    xmedia_isp_af_coring coring;
    xmedia_isp_af_ld      ld;
} xmedia_isp_af_h_param;
```

说明

宏定义如下：

```
#define XMEDIA_ISP_AF_IIR_EN_NUM      3
#define XMEDIA_ISP_AF_IIR_GAIN_NUM   4
#define XMEDIA_ISP_AF_IIR_FORWARD_COEF_NUM  9
#define XMEDIA_ISP_AF_IIR_BACKWARD_COEF_NUM 6
```

【成员】

成员名称	描述
iir_en	水平 iir 滤波器使能，三组滤波器开关。
bias_offset	iir 滤波器输出值截止值。
min_thd	iir 滤波器输出值截止阈值。
iir_gain	IIR 滤波器系数，用于控制 IIR 滤波器的频率响应。 取值范围：Gain:[-511, 511]。
forward_coeff	前馈滤波器系数，可设置三组。
backward_coeff	反馈滤波器强度控制，可控制三组
filter_mode	滤波器模式，0 求和 1 求峰值
coring	IIR 滤波器 Coring 模块配置。
ld	IIR 滤波器 Level Depend Gain 模块配置。

【注意事项】

无。

【相关数据类型及接口】

无

xmedia_isp_af_v_param

【说明】

AF 统计垂直滤波器 IIR 参数设置。

【定义】

```
typedef struct {
    xmedia_u16      min_thd;
    xmedia_s8      fir_gain[XMEDIA_ISP_AF_FIR_GAIN_NUM];
    xmedia_isp_af_coring coring;
    xmedia_isp_af_ld ld;
} xmedia_isp_af_v_param;
```

说明
宏定义如下：

```
#define XMEDIA_ISP_AF_FIR_GAIN_NUM 3
```

【成员】

成员名称	描述
min_thd	fir 滤波器输出值截止阈值
fir_gain	FIR 滤波器系数，取值范围：[-31, 31]。用于控制 FIR 滤波器的频率响应。
coring	FIR 滤波器 Coring 模块配置。
ld	FIR 滤波器 Level Depend Gain 模块配置。

【注意事项】

无。

【相关数据类型及接口】

无

xmedia_isp_fv_param

【说明】

AF 统计结果输出格式参数。

【定义】

```
typedef struct {
    xmedia_u16 shift_y;
    xmedia_u16 shift_fv1;
    xmedia_u16 shift_fv2;
    xmedia_u16 shift_hlt_cnt;
} xmedia_isp_fv_param;
```

【成员】

成员名称	描述
shift_y	亮度 Y 统计值移位寄存器值，取值范围：[0, 0xF]。
shift_fv1	水平 IIR 滤波值统计值移位寄存器值，取值范围：[0, 0xF]。
shift_fv2	垂直 FIR 滤波值统计值移位寄存器值，取值范围：[0, 0xF]。
shift_hlt_cnt	高亮计数器统计值移位寄存器值，取值范围：[0, 0xF]。

【注意事项】

当场景细节较多或噪声较多时，shift_v 值可能会较大溢出，这就需要对输出进行向右移位以保证输出在合理范围。用户根据自己的需求对相应的 shift 值进行调节。

【相关数据类型及接口】

无

xmedia_isp_af_statistics_cfg

【说明】

定义 AF 统计信息配置。

【定义】

```
typedef struct {
    xmedia_isp_af_cfg      config;
    xmedia_isp_af_h_param h_filter0;
    xmedia_isp_af_h_param h_filter1;
    xmedia_isp_af_v_param v_filter0;
    xmedia_isp_af_v_param v_filter1;
```

```

xmedia_isp_fv_param    fv_param;
} xmedia_isp_af_statistics_cfg;

```

【成员】

成员名称	描述
config	af 基础参数配置。
h_filter0	水平滤波器 0 参数配置。
h_filter1	水平滤波器 1 参数配置。
v_filter0	垂直滤波器 0 参数配置。
v_filter1	垂直滤波器 1 参数配置。
fv_param	fv 值移位配置。

【注意事项】

无。

【相关数据类型及接口】

无

xmedia_isp_ae_grid_info

【说明】

定义 AE 分区间统计信息的坐标信息。

【定义】

```

typedef struct {
    xmedia_u16 grid_y_pos[XMEDIA_ISP_AE_ZONE_ROW_MAX_NUM + 1];
    xmedia_u16 grid_x_pos[XMEDIA_ISP_AE_ZONE_COL_MAX_NUM + 1];
    xmedia_u8  status;
} xmedia_isp_ae_grid_info;

```

【成员】

成员名称	描述
grid_y_pos	AE 每个分区 Y 方向坐标值。
grid_x_pos	AE 每个分区 X 方向坐标值。
status	AE 统计信息坐标更新状态。

【注意事项】

无。

【相关数据类型及接口】

无

xmedia_isp_ae_statistics

【说明】

定义 AE 统计信息。

【定义】

```
typedef struct {
    xmedia_u32
    fe_zone_avg[XMEDIA_ISP_WDR_CHN_MAX_NUM][XMEDIA_ISP_AE_ZONE_ROW_MAX_NUM]
[XMEDIA_ISP_AE_ZONE_COL_MAX_NUM][XMEDIA_ISP_BAYER_PATTERN_NUM];
    xmedia_u32
    fe_global_avg[XMEDIA_ISP_WDR_CHN_MAX_NUM][XMEDIA_ISP_BAYER_PATTERN_NUM];
    xmedia_u32
    fe_hist_value[XMEDIA_ISP_WDR_CHN_MAX_NUM][XMEDIA_ISP_AE_HIST_NUM];
    xmedia_u32
    be_zone_avg[XMEDIA_ISP_AE_ZONE_ROW_MAX_NUM][XMEDIA_ISP_AE_ZONE_COL_MAX_
NUM] [XMEDIA_ISP_BAYER_PATTERN_NUM];
    xmedia_u32 be_global_avg[XMEDIA_ISP_BAYER_PATTERN_NUM];
    xmedia_u32 be_hist_value[XMEDIA_ISP_AE_HIST_NUM];
    xmedia_u32 fe_hist_weight_pixel_num[XMEDIA_ISP_WDR_CHN_MAX_NUM];
    xmedia_u32 be_hist_weight_pixel_num;
    xmedia_isp_ae_grid_info fe_grid_info;
    xmedia_isp_ae_grid_info be_grid_info;
} xmedia_isp_ae_statistics;
```

说明
宏定义如下：

【成员】

成员名称	描述
fe_zone_avg	FE 分区间统计平均值，按顺序分别表示 R,Gr,B,Gb 分量的平均值。
fe_global_avg	FE 全局统计平均值，按顺序分别表示 R,Gr,B,Gb 分量的平均值。
fe_hist_value	FE 全局 1024 段直方图信息
be_zone_avg	BE 分区间统计平均值，按顺序分别表示 R,Gr,B,Gb 分量的平均值。
be_global_avg	BE 全局统计平均值，按顺序分别表示 R,Gr,B,Gb 分量的平均值。
be_hist_value	BE 全局 1024 段直方图信息。
fe_hist_weight_pixel_num	fe 直方图带权重像素值个数。
be_hist_weight_pixel_num	be 直方图带权重像素值个数。
fe_grid_info	FE 分区间统计信息的坐标信息。
be_grid_info	BE 分区间统计信息的坐标信息。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_ae_blend_statistics

【说明】

定义拼接模式 AE 统计信息。

【定义】

```
typedef struct {  
    xmedia_u32
```

```

fe_zone_avg[XMEDIA_ISP_BLEND_MAX_NUM][XMEDIA_ISP_WDR_CHN_MAX_NUM]
[XMEDIA_ISP_WDR_CHN_MAX_NUM][XMEDIA_ISP_AE_ZONE_ROW_MAX_NUM][XMEDIA_IS
P_AE_ZONE_COL_MAX_NUM][XMEDIA_ISP_BAYER_PATTERN_NUM];
    xmedia_u32
fe_global_avg[XMEDIA_ISP_WDR_CHN_MAX_NUM][XMEDIA_ISP_BAYER_PATTERN_NUM];
    xmedia_u32
fe_hist_value[XMEDIA_ISP_WDR_CHN_MAX_NUM][XMEDIA_ISP_AE_HIST_NUM];
    xmedia_u32
be_zone_avg[XMEDIA_ISP_BLEND_MAX_NUM][XMEDIA_ISP_AE_ZONE_ROW_MAX_NUM][X
MEDIA_ISP_AE_ZONE_COL_MAX_NUM][XMEDIA_ISP_BAYER_PATTERN_NUM];
    xmedia_u32 be_global_avg[XMEDIA_ISP_BAYER_PATTERN_NUM];
    xmedia_u32 be_hist_value[XMEDIA_ISP_AE_HIST_NUM];
    xmedia_u32 fe_hist_weight_pixel_num[XMEDIA_ISP_WDR_CHN_MAX_NUM];
    xmedia_u32 be_hist_weight_pixel_num;
} xmedia_isp_ae_blend_statistics;

```



说明

宏定义如下：

```
#define XMEDIA_ISP_BLEND_MAX_NUM    4
```

【成员】

成员名称	描述
fe_zone_avg	FE 分区间统计平均值，按顺序分别表示 R,Gr,B,GB 分量的平均值。
fe_global_avg	FE 全局统计平均值，按顺序分别表示 R,Gr,B,Gb 分量的平均值。
fe_hist_value	FE 全局 1024 段直方图信息。
be_zone_avg	BE 分区间统计平均值，按顺序分别表示 R,Gr,B,Gb 分量的平均值。
be_global_avg	BE 全局统计平均值，按顺序分别表示 R,Gr,B,Gb 分量的平均值。
be_hist_value	BE 全局 1024 段直方图信息。
fe_hist_weight_pixel_num	fe 直方图带权重像素值个数。
be_hist_weight_pixel_num	be 直方图带权重像素值个数。

【注意事项】

无。

【相关数据类型及接口】

无

xmedia_isp_awb_grid_info

【说明】

awb 分区间统计信息的坐标信息。

【定义】

```
typedef struct {
    xmedia_u16 grid_y_pos[XMEDIA_ISP_AWB_ZONE_ROW_MAX_NUM + 1];
    xmedia_u16 grid_x_pos[XMEDIA_ISP_AWB_ZONE_COL_MAX_NUM + 1];
    xmedia_u8  status;
} xmedia_isp_awb_grid_info;
```

 说明
宏定义如下：

```
#define XMEDIA_ISP_AWB_ZONE_ROW_MAX_NUM    32
#define XMEDIA_ISP_AWB_ZONE_COL_MAX_NUM    32
```

【成员】

成员名称	描述
grid_y_pos	AWB 每个分区 Y 方向坐标值。
grid_x_pos	AWB 每个分区 X 方向坐标值。
status	AWB 统计信息坐标更新状态。

【注意事项】

无。

【相关数据类型及接口】

无

xmedia_isp_awb_statistics

【说明】

定义 AWB 统计信息。

【定义】

```
typedef struct {
    xmedia_u32 global_r;
    xmedia_u32 global_g;
    xmedia_u32 global_b;
    xmedia_u32 global_valid_count;
    xmedia_u32 zone_avg_r[XMEDIA_ISP_AWB_ZONE_NUM];
    xmedia_u32 zone_avg_g[XMEDIA_ISP_AWB_ZONE_NUM];
    xmedia_u32 zone_avg_b[XMEDIA_ISP_AWB_ZONE_NUM];
    xmedia_u32 zone_valid_count[XMEDIA_ISP_AWB_ZONE_NUM];
    xmedia_isp_awb_grid_info grid_info;
} xmedia_isp_awb_statistics;
```



说明

宏定义如下：

```
#define XMEDIA_ISP_AWB_ZONE_NUM (32*32)
```

【成员】

成员名称	描述
global_r	Bayer 域全局统计的 R 分量平均值，取值范围：[0x0, 0xFFFF]。
global_g	Bayer 域全局统计的 G 分量平均值，取值范围：[0x0, 0xFFFF]。
global_b	Bayer 域全局统计的 B 分量平均值，取值范围：[0x0, 0xFFFF]。
global_valid_count	全局统计的正方形灰色区域的像素个数。
zone_avg_r	Bayer 域分区间统计的 R 分量平均值，取值范围：[0x0, 0xFFFF]。
zone_avg_g	Bayer 域分区间统计的 G 分量平均值，取值范围：[0x0, 0xFFFF]。

zone_avg_b	Bayer 域分区间统计的 B 分量平均值，取值范围：[0x0, 0xFFFF]。
zone_valid_count	分区间统计的正方形灰色区域的像素个数。
grid_info	Awb 分区间统计信息的坐标信息。

【注意事项】

无。

【相关数据类型及接口】

无

xmedia_isp_awb_blend_statistics

【说明】

定义拼接模式 AWB 统计信息。

【定义】

```
typedef struct {
    xmedia_u32 zone_row[XMEDIA_ISP_BLEND_MAX_NUM];
    xmedia_u32 zone_col[XMEDIA_ISP_BLEND_MAX_NUM];
    xmedia_u32
zone_avg_r[XMEDIA_ISP_BLEND_MAX_NUM][XMEDIA_ISP_AWB_ZONE_NUM];
    xmedia_u32
zone_avg_g[XMEDIA_ISP_BLEND_MAX_NUM][XMEDIA_ISP_AWB_ZONE_NUM];
    xmedia_u32
zone_avg_b[XMEDIA_ISP_BLEND_MAX_NUM][XMEDIA_ISP_AWB_ZONE_NUM];
    xmedia_u32
zone_valid_count[XMEDIA_ISP_BLEND_MAX_NUM][XMEDIA_ISP_AWB_ZONE_NUM];
} xmedia_isp_awb_blend_statistics;
```

📖 说明

宏定义如下：

```
#define XMEDIA_ISP_BLEND_MAX_NUM    4
#define XMEDIA_ISP_AWB_ZONE_NUM    (32*32)
```

【成员】

成员名称	描述
zone_row	拼接统计分块行数，取值范围：[0x1,0x20]。
zone_col	拼接统计分块列数，取值范围：[0x1,0x100]。
zone_avg_r	Bayer 域分区间统计的 R 分量平均值，取值范围：[0x0, 0xFFFF]。
zone_avg_g	Bayer 域分区间统计的 G 分量平均值，取值范围：[0x0, 0xFFFF]。
zone_avg_b	Bayer 域分区间统计的 B 分量平均值，取值范围：[0x0, 0xFFFF]。
zone_valid_count	分区间统计的正方形灰色区域的像素个数。

【注意事项】

无。

【相关数据类型及接口】

无

xmedia_isp_af_zone

【说明】

AF 分块统计输出，分别输出分块 FV1 FV2 sum_y hlt_cnt。

【定义】

```
typedef struct {
    xmedia_u32 fv1_value;
    xmedia_u32 fv2_value;
    xmedia_u32 psduoy_value;
    xmedia_u32 hlt_cnt;
} xmedia_isp_af_zone;
```

【成员】

成员名称	描述
fv1_value	Fv1, 取值范围[0,0xFFFFFFFF]
fv2_value	Fv2, 取值范围[0,0xFFFFFFFF]
pseduoy_value	Sumy, 取值范围[0,0xFFFFFFFF]
hlt_cnt	高亮点个数,, 取值范围[0,0xFFFFFFFF]

【注意事项】

无

【相关数据类型及接口】

无

xmedia_ism_af_grid_info

【说明】

定义 AF 分区间统计信息的坐标信息。

【定义】

```
typedef struct {
    xmedia_u16 grid_y_pos[XMEDIA_ISP_AF_ZONE_ROW_MAX_NUM + 1];
    xmedia_u16 grid_x_pos[XMEDIA_ISP_AF_ZONE_COL_MAX_NUM + 1];
    xmedia_u8  status;
} xmedia_ism_af_grid_info;
```

说明
宏定义如下：

```
#define XMEDIA_ISP_AF_ZONE_ROW_MAX_NUM    15
#define XMEDIA_ISP_AF_ZONE_COL_MAX_NUM    18
```

【成员】

成员名称	描述
grid_y_pos	AF 每个分区 Y 方向坐标值。
grid_x_pos	AF 每个分区 X 方向坐标值。
status	AF 统计信息坐标更新状态。

【注意事项】

AF 分区间统计信息为 hwn d * vwnd 个均匀分区的统计信息。

【相关数据类型及接口】

无

xmedia_isp_fe_af_statistics

【说明】

定义 ISP AF 的 FE 统计信息。

【定义】

```
typedef struct {  
    xmedia_isp_af_zone  
    zone_metrics[XMEDIA_ISP_WDR_CHN_MAX_NUM][XMEDIA_ISP_AF_ZONE_ROW_MAX_NUM]  
    [XMEDIA_ISP_AF_ZONE_COL_MAX_NUM];  
    xmedia_isp_af_grid_info grid_info;  
} xmedia_isp_fe_af_statistics;
```

【成员】

成员名称	描述
zone_metrics	ISP AF FE 分块统计信息。
grid_info	ISP FE 的 AF 统计信息位置。

【注意事项】

暂未支持 FE 的 AF 统计。

【相关数据类型及接口】

无

xmedia_isp_be_af_statistics

【说明】

定义 ISP AF 的 BE 统计信息。

【定义】

```
typedef struct {  
    xmedia_isp_af_zone  
    zone_metrics[XMEDIA_ISP_AF_ZONE_ROW_MAX_NUM][XMEDIA_ISP_AF_ZONE_COL_MAX_NUM];  
    xmedia_isp_af_grid_info grid_info;  
} xmedia_isp_be_af_statistics;
```

【成员】

成员名称	描述
zone_metrics	ISP AF BE 分块统计信息。
grid_info	ISP BE 的 AF 统计信息位置。

【注意事项】

无。

【相关数据类型及接口】

无

xmedia_isp_af_statistics

【说明】

定义 ISP AF 统计信息。

【定义】

```
typedef struct {  
    xmedia_isp_fe_af_statistics fe_stat;  
    xmedia_isp_be_af_statistics be_stat;  
} xmedia_isp_af_statistics;
```

【成员】

成员名称	描述
fe_stat	AF 的 FE 统计信息配置。
be_stat	AF 的 BE 统计信息配置

【注意事项】

无。

【相关数据类型及接口】

无

仅限深圳市安远威视科技有限公司使用

10 Debug

10.1 概述

ISP 提供 debug MPI 供客户调用，以方便客户定位相关图像质量问题。

10.2 功能描述

Debug 提供记录 ISP 在运行过程中的状态信息，以方便记录在 ISP 运行过程中出现的异常状态，需要记录的帧数可以有用户自己指定。

10.3 API 参考

- [xmedia_isp_set_debug_mode](#): 设置 ISP 调试接口。
- [xmedia_isp_get_debug_mode](#): 获取 ISP 调试接口。
- [xmedia_ae_set_debug_info](#): 设置 AE 库 debug 功能输出。
- [xmedia_ae_set_debug_level](#): 设置 AE 库 debug 打印等级。

xmedia_isp_set_debug_mode

【描述】

设置 ISP 调试信息属性。

【语法】

```
xmedia_s32 xmedia_isp_set_debug_mode(xmedia_u32 pipe, const xmedia\_isp\_debug\_mode
*mode);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
isp_debug	调试信息。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_ae.h
- 库文件: libxmedia_isp.a

【注意】

- 调试信息的成员变量为 SYS，可以设置使能开关。
- 需要用户来分配内存以存储相应的调试信息，且需要将分配的内存地址作为输入传入。
- 暂未支持此接口。

【相关主题】

无

xmedia_isp_get_debug_mode

【描述】

获取 ISP 调试信息属性。

【语法】

```
xmedia_s32 xmedia_isp_get_debug_mode(xmedia_u32 pipe, xmedia_isp_debug_mode *mode);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
isp_debug	调试信息。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_ae.h
- 库文件: libxmedia_isp.a

【注意】

暂未支持此接口。

【举例】

无

xmedia_ae_set_debug_info

【描述】

设置 AE 库 debug 功能输出。

【语法】

```
xmedia_s32 xmedia_ae_set_debug_info (xmedia_u32 pipe, xmedia_isp_debug_info *debug_info);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
debug_info	ae 调试信息。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_ae.h
- 库文件: libxmedia_isp.a

【注意】

无

【举例】

无

xmedia_ae_set_debug_level

【描述】

设置 AE 库 debug 打印等级。

【语法】

```
xmedia_s32 xmedia_ae_set_debug_level(xmedia_u32 pipe, xmedia_u32 debug_level);
```

【参数】

参数名称	描述	输入/输出
pipe	pipe 号。	输入
debug_level	debug 打印等级。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件: xmedia_ae.h
- 库文件: libxmedia_isp.a

【注意】

无

10.4 数据类型

- xmedia_isp_debug_mode: 定义 ISP 调试信息属性。
- xmedia_isp_debug_info: 定义 AE 库的调试信息。

xmedia_isp_debug_mode

【说明】

定义 AE 库 debug 功能输出，打印信息输出到内存。

【定义】

```
typedef enum {
    XMEDIA_ISP_DEBUG_MODE_NONE = 0,
    XMEDIA_ISP_DEBUG_MODE_DEMOSAIC_DIR,
    XMEDIA_ISP_DEBUG_MODE_DRC_DETAIL,
    XMEDIA_ISP_DEBUG_MODE_WDR_LONG_WEIGHT,
    XMEDIA_ISP_DEBUG_MODE_WDR_LONG_20BIT,
    XMEDIA_ISP_DEBUG_MODE_WDR_LONG_12BIT,
    XMEDIA_ISP_DEBUG_MODE_WDR_SHORT_20BIT,
    XMEDIA_ISP_DEBUG_MODE_WDR_SHORT_12BIT,
    XMEDIA_ISP_DEBUG_MODE_STNR_TEXTURE,
    XMEDIA_ISP_DEBUG_MODE_STNR_EDGE,
    XMEDIA_ISP_DEBUG_MODE_STNR_MV,
    XMEDIA_ISP_DEBUG_MODE_STNR_EDGE_WEIGHT,
    XMEDIA_ISP_DEBUG_MODE_SHARPEN_TEXTURE,
```

```

XMEDIA_ISP_DEBUG_MODE_SHARPEN_EDGE,
XMEDIA_ISP_DEBUG_MODE_SHARPEN_MV,
XMEDIA_ISP_DEBUG_MODE_SHARPEN_EDGE_WEIGHT,
XMEDIA_ISP_DEBUG_MODE_MAX,
} xmedia_isp_debug_mode;

```

【成员】

成员名称	描述
XMEDIA_ISP_DEBUG_MODE_NONE	输出显示正常图像信息。
XMEDIA_ISP_DEBUG_MODE_DEMOSAIC_DIR	输出 demosaic 方向判断结果。
XMEDIA_ISP_DEBUG_MODE_DRC_DETAIL	输出 DRC 模块的细节层。
XMEDIA_ISP_DEBUG_MODE_WDR_LONG_WEIGHT	输出 wdr 模式的长帧融合权重。
XMEDIA_ISP_DEBUG_MODE_WDR_LONG_20BIT	输出 wdr 模式下 20bit 的长帧图像。
XMEDIA_ISP_DEBUG_MODE_WDR_LONG_12BIT	输出 wdr 模式下 12bit 的长帧图像。
XMEDIA_ISP_DEBUG_MODE_WDR_SHORT_20BIT	输出 wdr 模式下 20bit 的短帧图像。
XMEDIA_ISP_DEBUG_MODE_WDR_SHORT_12BIT	输出 wdr 模式下 12bit 的短帧图像。
XMEDIA_ISP_DEBUG_MODE_STNR_TEXTURE	sharpen 的 tex 层图像。
XMEDIA_ISP_DEBUG_MODE_STNR_EDGE	sharpen 的 edge 层图像。
XMEDIA_ISP_DEBUG_MODE_STNR_MV	sharpen 的运动图像。
XMEDIA_ISP_DEBUG_MODE_STNR_EDGE_WEIGHT	sharpen 细节层融

	合 edge 层的权重。
XMEDIA_ISP_DEBUG_MODE_SHARPEN_TEXTURE	sharpen 的 tex 层图像。
XMEDIA_ISP_DEBUG_MODE_SHARPEN_EDGE	sharpen 的 edge 层图像。
XMEDIA_ISP_DEBUG_MODE_SHARPEN_MV	sharpen 的运动图像。
XMEDIA_ISP_DEBUG_MODE_SHARPEN_EDGE_WEIGHT	sharpen 细节层融合 edge 层的权重。

【注意事项】

无

【相关数据类型及接口】

无

xmedia_isp_debug_info

【说明】

定义 ISP 调试信息属性。

【定义】

```
typedef struct {
    xmedia_bool debug_en; // rw; 1:enable debug, 0:disable debug
    xmedia_u64  phyaddr;  // rw; phy address of debug info
    xmedia_u32  depth;    // rw; depth of debug info - the frame number
} xmedia_isp_debug_info;
```

【成员】

成员名称	描述
debug_en	debug 功能的使能。
phyaddr	debug 信息的物理地址。
depth	debug 信息的帧数。

【注意事项】

无

【相关数据类型及接口】

无

仅限深圳市安远威视科技有限公司使用

11 错误码

ISP API 错误码如表 11-1 所示。

表11-1 ISP API 错误码

错误代码	宏定义	描述
0x80010001	XMEDIA_ERRCODE_INVALID_PARAM	输入参数无效
0x80010002	XMEDIA_ERRCODE_NULL_PTR	空指针错误
0x80010003	XMEDIA_ERRCODE_NOT_ENOUGH_MEMORY	申请内存失败
0x80010004	XMEDIA_ERRCODE_NOT_INIT	未初始化
0x80010005	XMEDIA_ERRCODE_NOT_SUPPORT	操作不支持
0x80010006	XMEDIA_ERRCODE_TIMEOUT	等待超时
0x80010007	XMEDIA_ERRCODE_BAD_ADDRESS	地址错误
0x80010008	XMEDIA_ERRCODE_OPEN_FAILED	打开设备节点失败
0x80010009	XMEDIA_ERRCODE_CLOSE_FAILED	关闭设备节点失败
0x8001000A	XMEDIA_ERRCODE_CREATE_FAILED	创建目标失败
0x8001000B	XMEDIA_ERRCODE_DESTROY_FAILED	销毁目标失败
0x8001000C	XMEDIA_ERRCODE_BUFFER_EMPTY	buffer 为空
0x8001000D	XMEDIA_ERRCODE_BUFFER_FULL	buffer 已满
0x8001000E	XMEDIA_ERRCODE_NO_BUFFER_FREE	无可用的 buffer
0x8001000F	XMEDIA_ERRCODE_NOT_PERMITTED	操作不允许
0x80010010	XMEDIA_ERRCODE_EXIST	目标已存在
0x80010011	XMEDIA_ERRCODE_NOT_EXIST	目标不存在

0x80010012	XMEDIA_ERRCODE_NOT_READY	目标未准备好
0x80010013	XMEDIA_ERRCODE_BUSY	设备或资源忙
0x80010014	XMEDIA_ERRCODE_COPY_DATA_ERROR	拷贝数据失败
0x80010015	XMEDIA_ERRCODE_TYPE_ERROR	通道类型超出支持的通道类型范围或未注册
0x80010016	XMEDIA_ERRCODE_CHN_FULL	注册的通道数超过最大值
0x80010017	XMEDIA_ERRCODE_INVALID_CHN_ID	无效通道 id
0x80010018	XMEDIA_ERRCODE_INVALID_DEV_ID	无效设备 id
0x80010019	XMEDIA_ERRCODE_INVALID_PIPE_ID	无效管道 id
0x8001001A	XMEDIA_ERRCODE_INVALID_GRP_ID	无效分组 id
0x8001001B	XMEDIA_ERRCODE_NOT_CONFIG	设备未配置
0x8001001C	XMEDIA_ERRCODE_NOT_ENABLE	设备未使能
0x8001001D	XMEDIA_ERRCODE_NOT_DISABLE	设备未失能
0x8001001E	XMEDIA_ERRCODE_NOT_BIND	数据源与数据接受者未绑定
0x8001001F	XMEDIA_ERRCODE_BINDED	数据源与数据接受者已绑定

12 Proc 调试信息说明

12.1 概述

调试信息采用了 proc 文件系统，可实时反映当前系统的运行状态，所记录的信息可供问题定位及分析时使用。proc 信息仅供参考，以实际输出为准。

【文件目录】

/proc/umap/isp

12.2 ISP

【调试信息】

```
Version:[SPC030 B020], [ISP V1.0 ], Release, Build Time[Sep 12 2025 18:19:52]

-----ISP PIPE[0] ISP INFO-----
-----
width| height| fps| wdr_mode| pixel_format| bayer_pattern| work_mode| slave_pipes|
blend_stat| blend_luma| follow_info|
3840| 2160| 30.00| linear| raw| bggr| master| -|
N| N| N/A|

-----ISP PIPE[0] SENSOR INFO-----
-----
sensor_type| max_width| max_height|max_wdr_mode| slave_mode_supported|
mipi_lanes_supported|
SC850SL| 3840| 2160| 2to1 wdr| N|
4|

width| height| wdr_format| max_fps| input_clock| output_intf| bit_width|
```

```

vcid|
      3840|      2160|      vc|      30.00|      24 MHz|      mipi csi|
12|      0|

-----ISP PIPE[0] VREG INFO-----
-----
      name|      id|      version|      size|
      isp|      0x584D00|      1003|      254400|
      awb|      0x584D01|      1001|      2184|
      ae|      0x584D02|      1001|      7280|
      public|      0x584D04|      1002|      16832|

-----ISP PIPE[0] TIME INFO-----
-----
      type |      count|      count(/s)|      max_count(/s)|      process_time|      max_process_time|
      isp_run|      9839|      30|      30|      2722|
16644|
      alg_run|      9840|      30|      30|      4497|
11168|
      alg_cfg|      9841|      30|      30|      824|
8072|
      proc|      328|      1|      1|      1631|
4690|

-----ISP PIPE[0] AWB INFO-----
-----
Version: [AWB V1.0 ],   Build Time[Aug 26 2025 20:10:44]

      shift|      zone_sel|multi_degree|      in_or_out|      light_bv|run_interval|
      64|      128|      0|      0|      32767|      2|

      awb_bypass|      awb_type|      ccm_en|      ccm_type|      manu_en|      sat_type|
      0|      0|      1|      0|      1|      0|

      co_temp|      sat|      r_gain|      gr_gain|      gb_gain|      b_gain|
      5909|      128|      527|      256|      256|      492|

      ccm00|      ccm01|      ccm02|      ccm10|      ccm11|      ccm12|
ccm20|      ccm21|      ccm22|
      437|      32963|      14|      32869|      408|      32819|
32769|      32993|      482|

-----ISP PIPE[0] AE INFO-----

```

Version: [AE V1.1], Build Time[Aug 26 2025 20:10:42]

sys_gain	int_time_sf	int_time	exp_value	ori_avg	exp_ratio	error
2199	0	33259	308597	14990	0	-

664|

ae_target	speed	fps	real_fps	black_delay	white_delay
14336	64	30.00	30.00	0	0

wdr_mode	anti_flick	slow_mode	gain_thr	manual_exp_en	manual_exp_val
0	0	1	0	0	1024000

node_id	int_time	sys_gain
0	30	1024
1	33274	1024
2	33274	33554432

again	dgain	again_sf	dgain_sf	isp_dgain	isp_dgain_sf	iso
iso_sf	2176	1024	1024	1024	258	256

214| 100|

ae_inter_val	int_time	wdr_gain[0]	wdr_gain[1]	hmax	vmax
1	33259	256	256	14814	2250

bypass_en	min_expratio	max_expratio	op_type
0	128	16384	auto

-----ISP PIPE[0] BLC INFO-----

mode	b1c_0	b1c_1	b1c_2	b1c_3
auto	271	271	271	271

-----ISP PIPE[0] AINR INFO-----

enable	active	iso_pre	a	b	thd_high	thd_low
0	0	0	0	0	8000	6400

coef0	coef1	coef2
coef3	coef4	
0.0003627490000000	-0.0040428115000000	0.0000000507000000
0.0000000578000000	0.0014511087000000	-

-----ISP PIPE[0] CROSSTALK INFO-----						
enable	mode	slope	threshold	np_offset		
1	auto	0	100	0		
-----ISP PIPE[0] CSC INFO-----						
enable	color_gamut	hue	luma	contrast	saturation	
limited_range_en						
1	bt709	50	50	50	50	
0						
-----ISP PIPE[0] DH INFO-----						
enable	userlut_en	mode	incr_coef	decr_coef		
1	0	auto	0	0		
-----ISP PIPE[0] DGN INFO-----						
dgn0_en	dgn1_en	dgn2_en	dgn3_en			
0	1	1	1			
dgn0[0][0]	dgn0[0][1]	dgn0[0][2]	dgn0[0][3]	dgn0[1][0]	dgn0[1][1]	dgn0[1][2]
dgn0[1][3]						
256	256	256	256	256	256	256
256	256					
dgn1[0][0]	dgn1[0][1]	dgn1[0][2]	dgn1[0][3]	dgn1[1][0]	dgn1[1][1]	dgn1[1][2]
dgn1[1][3]						
258	258	258	258	258	256	256
256	256					
dgn2						
275						
dgn3[0]	dgn3[1]	dgn3[2]	dgn3[3]			
527	256	256	492			
-----ISP PIPE[0] AAC FE-AE INFO-----						
global_avg_en	zone_avg_en	hist_en	pos_switch	hist_mode		
hist_four_mode	h_zone_num	v_zone_num				

single_color	1	24	1	16	dgn2	weight
-----ISP PIPE[0] AAC BE-AE INFO-----						
global_avg_en	zone_avg_en	hist_en	pos_switch	hist_mode		
hist_four_mode	h_zone_num	v_zone_num				
single_color	1	24	1	16	dgn2	weight
-----ISP PIPE[0] AAC AWB INFO-----						
global_avg_en	zone_avg_en	pos_switch	h_zone_num	v_zone_num		
	1	1	dgn2	32	32	
-----ISP PIPE[0] AAC AF INFO-----						
pdaf_stat_en	cdaf_stat_en	pos_switch	h_zone_num	v_zone_num		
h_size	v_size	hlt_thd				
	0	1	csc	18	15	
3840	2160	3855				
-----ISP PIPE[0] BNR INFO-----						
enable	mode	ainr_pos_switch	tnr_enable	tnr_ref_buff		
1	auto	off	1	1		
-----ISP PIPE[0] DMS INFO-----						
enable	mode	mf_de_str	mf_de_osh_str	mf_de_ush_str	hf_de_str	
1	auto	64	128	128	128	
-----ISP PIPE[0] DPC INFO-----						
dynamic_enable	mode	sup_twinkle_en	soft_thd	soft_slope		
1	auto	0	0	0		
static_enable	pd_enable	bright_count	dark_count	show		
0	0	0	0	0		
-----ISP PIPE[0] DRC INFO-----						
enable	ltmc_en	dering_en	detail_enhance_en	tm_type		

mode	str				
0	1	1	1	asymmetry	
manual	256				

-----ISP PIPE[0] EXPANDER INFO-----

enable	bit_in	bit_out	knee_point_num
0	0	0	0

-----ISP PIPE[0] FPN INFO-----

enable	fe_active	be_active	bin_phy_addr(Hex)
0	0	0	0

-----ISP PIPE[0] GAMMA INFO-----

enable	mode	curve_type
1	auto	user_define

-----ISP PIPE[0] GCAC INFO-----

enable	ver_coord	hor_coord
1	719	1279

-----ISP PIPE[0] HLC INFO-----

enable	luma_thr	luma_target
0	940	10

-----ISP PIPE[0] LCE INFO-----

enable	mode
1	auto

-----ISP PIPE[0] MLSC INFO-----

enable	mesh_str
1	256

-----ISP PIPE[0] RC INFO-----

enable	corr_x	corr_y	radius
0	1920	1080	1080

-----ISP PIPE[0] RGBIR INFO-----

enable	in_pattern	out_pattern
0	bggr	rggb

-----ISP PIPE[0] RLSC INFO-----

enable
0

-----ISP PIPE[0] WDR INFO-----

enable	wbgain_pos
0	dg1

-----ISP PIPE[0] CA INFO-----

enable	mode	sat_str
1	auto	2

-----ISP PIPE[0] IRQ INFO-----

irq	count	count(/s)	max_count(/s)	process_time	max_process_time
pt_fs	0	0	0	0	0
pt_fe	0	0	0	0	0
fe_fs	9870	30	30	35	2528
fe_fe	9869	30	30	32	44
be_fs	0	0	0	0	0
be_fe	9868	30	30	115	138

-----ISP PIPE[0] ALG INFO-----

queue	node_num	cur_free	cur_busy	min_free(get)	max_busy(put)	repeat_total
repeat_max	unused_num					
isp_fe_buff	4	2	2	2	3	1
1	0					
isp_be_buff	5	3	2	3	3	0
0	0					
calc_frame_num	missed_run_num	discard_frame_num	overwrite_num	fe_get_num		
fe_put_num	be_get_num	be_put_num				
	9870	0		0	0	9871

9869| 9869| 9868|

buff_name	buff_num	buff_size	total_size
isp_fe_buff	4	0x000020E0	0x00008380
isp_be_buff	5	0x00009740	0x0002F440
frame_info_buff	1	0x00000058	0x00000058

-----ISP PIPE[0] STAT INFO-----

queue	node_num	cur_free	cur_busy	min_free(get)	max_busy(put)	repeat_total
repeat_max	unused_num					
isp_fe_stat	2	1	1	1	1	0
0	0					
isp_be_stat	2	1	1	1	1	1
1	0					
isp_fe_attr	5	3	2	4	2	0
0	0					
isp_be_attr	5	4	1	5	1	0
0	0					

fe_format_cnt	be_format_cnt
9869	9868

fe_3a_buf_single_size	fe_3a_buf_num	fe_3a_buf_total_size
0x00002900	4	0x0000A400
be_3a_buf_single_size	be_3a_buf_num	be_3a_buf_total_size
0x00008900	2	0x00011200
be_lce_buf_single_size	be_lce_buf_num	be_lce_buf_total_size
0x00000D00	2	0x00001A00

be_fifo_num
1

fe_buff_single_size	fe_buff_num	fe_buff_total_size	be_buff_single_size	be_buff_num
be_buff_total_size				
0x00005090	2	0x0000A120	0x00008708	
0x00010E10				

fe_attr_single_size	fe_attr_num	fe_attr_total_size	be_attr_single_size	be_attr_num
be_attr_total_size				
0x00000070	5	0x00000230	0x00000160	
0x000006E0				

-----ISP PIPE[0] END-----

参数		描述
ISP INFO	width	图像的宽度。
	height	图像的高度。
	fps	图像的帧率。
	wdr_mode	WDR 工作模式。
	pixel_format	图像的像素格式。
	bayer_pattern	图像的 bayer 格式。
	work_mode	isp 的工作模式。
	master_pipes /slave_pipes	多通路场景下，与 isp 的工作模式相关： master_pipes: 工作模式为 slave 模式的从 pipe 所绑定的主 pipe 号； slave pipes: 工作模式为 master 模式的主 pipe 所绑定的从 pipe 号。
	blend_stat	融合统计信息开关。
	blend_luma	亮度统计信息开关。
SENSOR INFO	follow_info	多通路场景下，从 pipe 依赖主 pipe 的算法参数信息，1 个 bit 位代表一个算法模块。
	sensor_type	Sensor 型号。
	max_width	sensor 图像的最大宽度。
	max_height	sensor 图像的最大高度。
	max_wdr_mode	支持最大的 wdr 帧数。

	slave_mode_supported	是否支持从模式。
	mipi_lanes_supported	sensor 支持的最大 mipi lanes 数。
	width	Sensor 当前输出的宽度。
	height	Sensor 当前输出的高度。
	wdr_format	Sensor 当前所支持的 wdr 模式。
	max_fps	sensor 当前所支持的最大帧率。
	input_clock	sensor 的输入时钟频率。
	output_intf	sensor 输出接口的协议。
	bit_width	sensor 当前所支持的最大 bit 位宽。
	vcid	WDR 模式下, 使用 VC 模式传输 sensor 数据的 vc 号。
GROUP INFO	group	Group 功能组 group id。
	type	Group 功能组类型。
	pipe_num	Group 功能组 pipe 数量。
	pipe_id	Group 功能组 pipe 号。
	trig_num	Group 功能组类型为 HOST_SLAVE 时, 功能组内 sensor 配置的 trig 数量; 功能组类型为 MASTER_SLAVE 时, trig 数量为 0。
	trig_id	Group 功能组类型为 HOST_SLAVE 时, 功能组内 sensor 配置的 trig 设备号; 功能组类型为 MASTER_SLAVE 时, trig 数量为 N/A。
VREG INFO	name	表示 vreg 模块的名称

	id	表示 vreg 模块的 id 号。
	version	表示 vreg 模块的版本号。
	size	表示 vreg 模块所占内存的大小 (单位为字节)。
TIME INFO	type	统计类型。
	count	统计次数。
	count(s)	次数/秒。
	max count(s)	最大次数/秒。
	process time	处理时间。
	max process time	最大处理时间。
AWB INFO	shift	普朗克曲线的两条外扩曲线半径 (白点范围框选)。
	zone_sel	普朗克曲线的两条外扩曲线的偏移量微调。
	multi_degree	混合色温程度。
	in_or_out	室内外状态判断。
	light_bv	当前环境曝光量 bv 值。
	run_interval	awb 运行频率(单位: 帧 / 次)
	awb_bypass	awb 是否 bypass。
	awb_type	awb 自动手动类型。
	ccm_en	ccm 是否使能。
	ccm_type	ccm 自动手动类型。
	manu_en	手动 ccm 时是否使能 sat 饱和度调整。
	sat_type	饱和度自动手动类型。
	co_temp	Color temperature

	sat	饱和度强度
	r_gain	Rgain
	gr_gain	Grgain
	gb_gain	Gbgain
	b_gain	Bgain
	ccm00	ccm[0]
	ccm01	ccm[1]
	ccm02	ccm[2]
	ccm10	ccm[3]
	ccm11	ccm[4]
	ccm12	ccm[5]
	ccm20	ccm[6]
	ccm21	ccm[7]
	ccm22	ccm[8]
AE INFO	sys_gain	当前系统增益，10bit 精度。
	int_time_sf	linear 模式下表示曝光时间，wdr 模式下表示短帧曝光时间，单位 us。
	int_time	wdr 模式下表示长帧曝光时间，单位 us。
	exp_value	曝光量，为曝光行数（line）与系统增益（6bit 精度）乘积。
	ori_avg	不带亮度权重的画面平均亮度值，16bit 数据上统计。
	exp_ratio	wdr 模式下曝光比。
	error	曝光目标亮度值与加权亮度值差异，error 小于 tolerance 时 AE 收敛。
	ae_target	AE 曝光目标亮度值。
	speed	AE 收敛速度。

fps	当前基准帧率。
real_fps	当前实际帧率，非 AE 自动降帧场景下等于基准帧率。
black_delay	图像亮度小于目标亮度时 AE 延迟生效帧数。
white_delay	图像亮度大于目标亮度时 AE 延迟生效帧数。
wdr_mode	当前 WDR 模式。
anti_flick	Antiflicker 开关。
slow_mode	慢快门模式选择。
gain_thr	AE 自动降帧阈值。
manual_exp_en	手动曝光量功能开关。
manual_exp_val	手动曝光量数值。
node_id	AE Route 节点序号。
int_time	Route 节点曝光时间。
sys_gain	Route 节点系统增益。
again	RouteEx 节点 sensor 模拟增益。
dgain	RouteEx 节点 sensor 数字增益。
isp_dgain	RouteEx 节点 ISP 数字增益。
again_sf	RouteEx 节点短帧 sensor 模拟增益。
dgain_sf	RouteEx 节点短帧 sensor 数字增益。
isp_dgain_sf	RouteEx 节点短帧 ISP 数字增益。
iso	系统 ISO 值。
iso_sf	短帧 ISO 值。

	ae_inter_val	AE 运行间隔。
	int_time	linear 模式下表示曝光时间, wdr 模式下表示短帧曝光时间, 单位 us。
	wdr_gain[0]	WDR 融合前长帧增益。
	wdr_gain[1]	WDR 融合前短帧增益。
	hmax	Sensor 单行曝光时间。
	vmax	Sensor 最大曝光行数。
	bypass_en	AE 是否 bypass。
	min_expratio	WDR 模式下, 支持的最小曝光比, 64 为 1 倍。
	max_expratio	WDR 模式下, 支持的最大曝光比, 64 为 1 倍。
	op_type	AE 手动 / 自动模式。
ANTI_FALSE_COLOR INFO	enable	AFC 模块使能开关。
	fcr_str	去伪彩强度。
BLC INFO	mode	BLC 操作模式。
	bld_0	R 通道 BLC 值。
	bld_1	Gr 通道 BLC 值。
	bld_2	Gb 通道 BLC 值。
	bld_3	B 通道 BLC 值。
AINR INFO	enable	AINR 使能开关。
	active	AINR 生效标志位。
	iso	当前场景的 ISO 值。
	a	Poisson 降噪强度。
	b	Gaussian 降噪强度。

	thd_high	AINR 开关控制高阈值。
	thd_low	AINR 开关控制低阈值。
	coef0	Noise profile 标定系数 0。
	coef1	Noise profile 标定系数 1。
	coef2	Noise profile 标定系数 2。
	coef3	Noise profile 标定系数 3。
CROSSTALK INFO	enable	Crosstalk 使能开关。
	mode	Crosstalk 操作模式。
	slope	Crosstalk 门限之上的强度。
	threshold	Crosstalk 门限强度。
	np_offset	noise profile 值。
CSC INFO	enable	CSC 使能开关。
	color_gamut	色彩空间类型。
	hue	目标 YUV 空间色调值。
	luma	目标 YUV 空间亮度值。
	contrast	目标 YUV 空间对比度值。
	saturation	目标 YUV 空间饱和度值。
	limited_range_en	目标 YUV 空间动态范围类型。
DH INFO	enable	使能 Dehaze 功能。
	userlut_en	用户自定义去雾曲线使能。
	mode	Dehaze 操作模式。
	incr_coef	时域滤波递增系数。
	decr_coef	时域滤波递减系数。
DGN INFO	dgn0_en	DGN0 使能开关。
	dgn1_en	DGN1 使能开关。

	dgn2_en	DGN2 使能开关。
	dgn3_en	DGN3 使能开关。
	dgn0[0][0]	DGN0 chn0 R 通道增益值。
	dgn0[0][1]	DGN0 chn0 Gr 通道增益值。
	dgn0[0][2]	DGN0 chn0 Gb 通道增益值。
	dgn0[0][3]	DGN0 chn0 B 通道增益值。
	dgn0[1][0]	DGN0 chn1 R 通道增益值。
	dgn0[1][1]	DGN0 chn1 Gr 通道增益值。
	dgn0[1][2]	DGN0 chn1 Gb 通道增益值。
	dgn0[1][3]	DGN0 chn1 B 通道增益值。
	dgn1[0][0]	DGN1 chn0 R 通道增益值。
	dgn1[0][1]	DGN1 chn0 Gr 通道增益值。
	dgn1[0][2]	DGN1 chn0 Gb 通道增益值。
	dgn1[0][3]	DGN1 chn0 B 通道增益值。
	dgn1[1][0]	DGN1 chn1 R 通道增益值。
	dgn1[1][1]	DGN1 chn1 Gr 通道增益值。
	dgn1[1][2]	DGN1 chn1 Gb 通道增益值。
	dgn1[1][3]	DGN1 chn1 B 通道增益值。
	dgn2	DGN2 增益值。
	dgn3[0]	DGN3 R 通道增益值。
	dgn3[1]	DGN3 Gr 通道增益值。
	dgn3[2]	DGN3 Gb 通道增益值。
	dgn3[3]	DGN3 B 通道增益值。
AAC FE-AE INFO	global_avg_en	全局均值使能。
	zone_avg_en	分块均值使能。

	hist_en	直方图使能。
	pos_switch	Fe ae 位置选择。
	hist_mode	直方图模式。
	hist_four_mode	直方图统计模式。
	h_zone_num	水平方向分块个数。
	v_zone_num	垂直方向分块个数。
AAC BE-AE INFO	global_avg_en	全局均值使能。
	zone_avg_en	分块均值使能。
	hist_en	直方图使能。
	pos_switch	Be ae 位置选择。
	hist_mode	直方图模式。
	hist_four_mode	直方图统计模式。
	h_zone_num	水平方向分块个数。
	v_zone_num	垂直方向分块个数。
AAC AWB INFO	global_avg_en	全局均值使能。
	zone_avg_end	分块均值使能。
	pos_switch	WB 统计模块位置。
	h_zone_num	WB 统计水平方向分块个数。
	v_zone_num	WB 统计垂直方向分块个数。
AAC AF INFO	pdaf_stat_en	pdaf 统计使能。
	cdaf_stat_en	cdaf 统计使能。
	pos_switch	AF 统计信息位置。
	h_zone_num	AF 统计水平方向窗口个数。
	v_zone_num	AF 统计垂直方向窗口个数。
	h_size	AF 统计输入图像宽度。

	v_size	AF 统计输入图像高度。
	hlt_thd	高亮点最低亮度阈值。
BNR INFO	enable	BNR 使能开关。
	mode	BNR 操模式。
	ainr_pos_switch	AINR 结果的融合位置。
	tnr_enable	B3D 去噪使能开关。
	tnr_ref_buff	B3D 参考帧 buffer 分配标志位。
DMS INFO	enable	Demosaic 模块使能。
	mode	Demosaic 操作模式。
	mf_de_str	中频增强强度。
	mf_de_osh_str	细节增强 overshoot 控制强度。
	mf_de_ush_str	细节增强 undershoot 控制强度。
	hf_de_str	高频增强强度。
DPC INFO	dynamic_enable	动态坏点校正功能使能。
	mode	动态坏点校正操作模式。
	sup_twinkle_en	闪烁抑制使能。
	soft_thd	闪烁抑制的阈值。
	soft_slope	闪烁抑制模块中，控制偏差小于 soft_thr 的像素点的校正情况。值越大，参与校正处理的像素点越多，图像会越模糊。
	static_enable	静态坏点标定使能。
	pd_enable	pdaf 点的校正使能。
	bright_count	亮坏点个数。
	dark_count	暗坏点个数。
	show	静态坏点显示使能。

DRC INFO	enable	DRC 使能开关。
	ltmc_en	LTMC 开关。
	dering_en	Dering 功能开关
	detail_enhance_en	细节增强开关。
	tm_type	DRC 曲线类型。
	mode	DRC 操作模式。
	str	drc 强度值。
EXPANDER INFO	enable	expander 模块使能模块。
	bit_in	输入数据位宽。
	bit_out	输出数据位宽。
	knee_point_num	拐点坐标的数目。
FPN INFO	enable	FPN 校正功能使能开关。
	fe_enable	FE FPN 校正功能使能开关。
	be_enable	BE FPN 校正功能使能开关。
	bin_phy_addr(Hex)	FPN 黑帧物理首地址。
GAMMA INFO	enable	Gamma 校正功能使能。
	mode	Gamma 操作模式。
	curve_type	Gamma 曲线类型。
GCAC INFO	enable	横向色差校正使能。
	ver_coord	光学中心纵坐标值。
	hor_coord	光学中心横坐标。
LCAC INFO	enable	LCAC 模块使能。
	purple_upper_limit	去除紫边颜色上限。
	purple_lower_limit	去除紫边颜色下限。
	de_purple_cr_str	CR 通道去紫边强度。

	de_purple_cb_str	CB 通道去紫边强度。
HLC INFO	enable	HLC 高光遮蔽的使能开关。
	luma_thr	高光遮蔽的阈值。
	luma_target	高光遮蔽拉灰的目标值。
LCE INFO	enable	LCE 使能位。
	mode	LCE 操作模式。
MLSC INFO	enable	Mesh Shading 使能位。
	mesh_str	Mesh Shading 标定后的强度。
RC INFO	enable	Radial Crop 使能。
	corr_x	中心点的坐标配置 x。
	corr_y	中心点坐标配置 y。
	radius	拉黑的半径。
RGBIR INFO	enable	RGBIR 使能开关。
	in_pattern	RGBIR 输入 bayer pattern。
	out_pattern	RGBIR 输出 bayer pattern。
RLSC INFO	enable	RLSC 使能开关。
WDR INFO	enable	WDR 使能开关。
	wbgain_pos	白平衡增益在 ISP 的位置。
CA INFO	enable	CA 使能开关。
	mode	CA 操作模式。
	sat_str	yuv 域饱和度调整全局强度。
SHARPEN INFO	enable	Sharpen 模块开关。
	ysh_detail_mode	Sharpen 细节融合模式。
	ysh_edge_str	edge 全局锐化强度。
	ysh_tex_str	texture 全局锐化强度。

	ysh_over_shoot_str	全局白边控制强度。
	ysh_under_shoot_str	全局黑边控制强度。
STNR INFO	stnr_enable	stnr 总开关。
	stnr_mode	选择 YNR/TNR/YSH 顺序的模式。
	ynr_enable	亮度域空域去噪使能。
	denoise_str_mode	去噪基础强度是否参考运动区模式选择。
	v_denoise_str	控制值域核去噪强度。
	nr_addback_en	噪声回叠开关。
	nr_addback_str	噪声回叠强度。
	cnr_enable	色度域空域去噪使能。
	coarse_denoise_str	去色噪强度（粗调）。
	mot_bld_str	运动区域融合强度。
	mot_bld_thd_low	运动区域融合低阈值。
	mot_bld_thd_high	运动区域融合高阈值。
	tnr_enable	时域去噪使能。
	md_win_size	小图检测运动时所用的窗口大小。
	bgm_str	小图时域降噪强度。
	str	Y 通道时域降噪强度。
	uv_str	UV 通道时域降噪强度。
	md_thd_low	运动检测阈值下限。
	md_thd_high	运动检测阈值上限。
	border_thd	小图边界处理阈值。
	border_md_scl	小图边界处理强度。

IRQ INFO	irq	中断类型。 pt_fs: pt 帧起始中断; pt_fe: pt 帧完成中断; fe_fs: fe 帧起始中断; fe_fe: fe 帧完成中断; be_fs: be 帧起始中断; be_fe: be 帧完成中断。
	count	次数。
	count(/s)	次数/秒。
	max count(/s)	最大次数/秒。
	process time	处理时间。
	max process time	最大处理时间。
ALG INFO	queue	队列名。
	node_num	节点数。
	cur_free	当前空闲节点数。
	cur_busy	当前正在使用的节点数。
	min_free(get)	最小空闲节点数（在获取空闲节点前统计）。
	max_busy(put)	最大使用节点数（在放入使用节点后统计）。
	repeat_total	重复使用配置的总次数。
	repeat_max	重复使用同一套配置的最大次数。
	unused_num	队列中未被使用的配置次数。
	calc_frame_num	计算配置的帧计数。
	missed_run_num	ISP run 线程未被唤醒的次数。
	discard_frame_num	丢帧的帧计数。

	overwrite_num	配置未使用的次数。
	fe_get_num	fe 配置获取次数。
	fe_put_num	fe 配置归还次数。
	be_get_num	be 配置获取次数。
	be_put_num	be 配置归还次数。
	buff_name	buffer 名字。
	buff_num	buffer 数量。
	buff_size	buffer 大小, 单位: 字节。
	total_size	总大小, 单位: 字节。
	isp_fe_buff	fe buffer 物理首地址。
	isp_be_buff	be buffer 物理首地址。
	frame_info_buff	frame info buffer 物理首地址。
STAT INFO	queue	队列名。
	node_num	节点数。
	cur_free	当前空闲节点数。
	cur_busy	当前正在使用的节点数。
	min_free(get)	最小空闲节点数 (在获取空闲节点前统计)。
	max_busy(put)	最大使用节点数 (在放入使用节点后统计)。
	repeat_total	重复使用配置的总次数。
	repeat_max	重复使用同一套配置的最大次数。
	unused_num	队列中未被使用的配置次数。
	fe_format_cnt	fe 统计信息格式化次数。
	be_format_cnt	be 统计信息格式化次数。

	fe_3a_buf_single_size	fe 3a 统计信息 buffer 单个大小，单位：字节。
	fe_3a_buf_num	fe 3a 统计信息 buffer 数量。
	fe_3a_buf_total_size	fe 3a 统计信息 buffer 总大小，单位：字节。
	be_3a_buf_single_size	be 3a 统计信息 buffer 单个大小，单位：字节。
	be_3a_buf_num	be 3a 统计信息 buffer 数量。
	be_3a_buf_total_size	be 3a 统计信息 buffer 总大小，单位：字节。
	be_lce_buf_single_size	be lce 统计信息 buffer 单个大小，单位：字节。
	be_lce_buf_num	be lce 统计信息 buffer 数量。
	be_lce_buf_total_size	be lce 统计信息 buffer 总大小，单位：字节。
	be_fifo_num	be 待格式化的统计信息 buffer 数。
	fe_buff_single_size	fe 统计信息 buffer 单个大小，单位：字节。
	fe_buff_num	fe 统计信息 buffer 数量。
	fe_buff_total_size	fe 统计信息 buffer 总大小，单位：字节。
	be_buff_single_size	be 统计信息 buffer 单个大小，单位：字节。
	be_buff_num	be 统计信息 buffer 数量。
	be_buff_total_size	be 统计信息 buffer 总大小，单位：字节。
	fe_attr_single_size	fe 统计信息配置 buffer 单个大小，单位：字节。

	fe_attr_num	fe 统计信息配置 buffer 数量。
	fe_attr_total_size	fe 统计信息配置 buffer 总大小， 单位：字节。
	be_attr_single_size	be 统计信息配置 buffer 单个大小，单位：字节。
	be_attr_num	be 统计信息配置 buffer 数量。
	be_attr_total_size	be 统计信息配置 buffer 总大小， 单位：字节。

13

3A 开发指南概述

13.1 概述

ISP 3A 版本依赖于相应的 SDK 大版本，通过一系列数字图像处理算法完成对数字图像的效果处理。主要包含 firmware 框架及 XMEDIA_3A 库，firmware 提供算法的基本框架，处理统计信息，驱动数字图像处理算法，并包含坏点校正、去噪、色彩增强、镜头阴影校正等处理。3A 库以注册的方式，添加到 Firmware 中，完成曝光、白平衡、色彩还原等处理。

13.2 功能描述

ISP 的 Firmware 架构及 Firmware 工作流程及文件组织结构请参考本文档 1.2 “功能描述”小节，本小节不再赘述。

14

3A 开发使用者指南

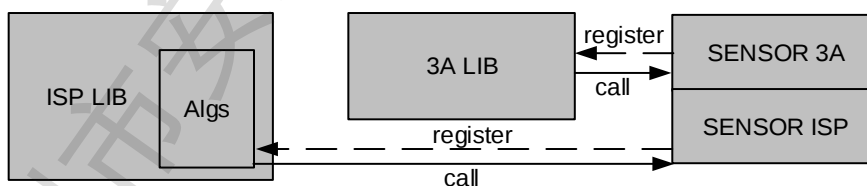
14.1 软件流程

ISP 软件流程请参考第一章 1.2.4 “软件流程” 小节，本小节不再赘述。

14.2 Sensor 对接

sensor 库主要是为了提供差异化的 sensor 适配，里面的内容可以分为两部分：sensor 向 ISP 库注册的差异化适配，这些差异化适配主要由 firmware 中的基础算法单元决定；sensor 向 XMEDIA_3A 库注册的差异化适配。sensor 的适配包括算法的初始化默认值及 sensor 控制接口，sensor 的适配是通过接口回调的形式注册给 ISP 库和 3A 算法库。图 14-1 表示了 sensor 与 ISP 库和 3A 算法库之间的关系。

图14-1 Sensor 适配示意图



关于 sensor 注册 ISP 库，sensor 注册 3A 算法库部分详细请参考《sensor 调试指南》。

15 开发者指南

15.1 概述

用户可以基于 firmware 框架开发定制 3A 库，并注册到 firmware 中，firmware 将会在中断的驱动下，获取每帧的统计信息，运行算法库，配置 ISP 寄存器。

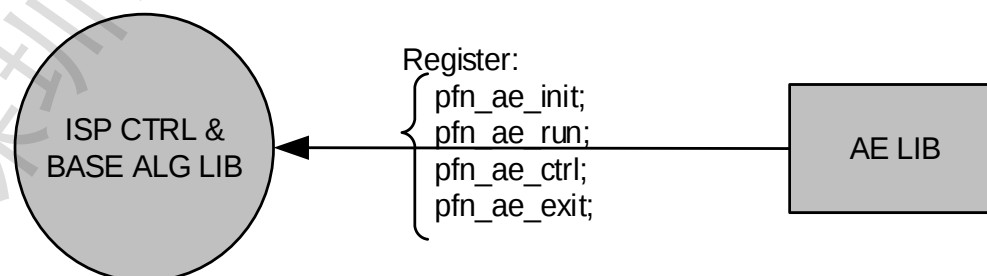
sensor 注册到 ISP 库，以实现差异化适配 Firmware 中的基础算法单元的部分，仍需要参考使用者指南部分的内容实现，以保证 firmware 中的坏点校正、去噪、色彩增强、镜头阴影校正等处理算法正常运行。sensor 注册到 3A 算法库的部分，请用户根据自己开发定制的 3A 库，自行定义数据结构，实现 Sensor 曝光控制等。

如果用户只是使用 XMEDIA_3A 算法库，并不自己开发 3A 算法库，可以忽略此章节内容。

15.2 AE 算法注册 ISP 库

AE 算法注册 ISP 库调用 `xmedia_isp_register_ae_lib`，如图 15-1 所示：

图15-1 Sensor 适配示意图



XMEDIA_AE 算法实现了一个 xmedia_ae_register 的注册函数，在这个函数中调用 ISP 提供的 xmedia_isp_register_ae_lib 回调接口，用户调用注册函数以实现向 ISP 注册 AE 算法，示例如下：

【举例】

```
/* 实现注册函数 */
xmedia_u32 pipe = 0;
xmedia_isp_ae_register_info ae_register;
xmedia_s32 ret = XMEDIA_SUCCESS;
strcpy(ae_register.alg_name, AE_LIB_NAME);
ae_register.alg_id      = pipe;
/* 调用钩子函数 */

ae_register.pfn_ae_init = ae_init;
ae_register.pfn_ae_run  = ae_run;
ae_register.pfn_ae_ctrl = ae_ctrl;
ae_register.pfn_ae_exit = ae_exit;
ret = xmedia_isp_register_ae_lib(pipe, &ae_register, &(ae_ctx->sns_register));
AE_CHECK_RET_RETURN(ret, "xmedia_isp_register_ae_lib failed!");
```

用户需要在自开发定制的 AE 库中实现以下回调函数：

表15-1 AE 算法向 ISP 库注册的回调函数

成员名称	描述
pfn_ae_init	初始化 AE 的回调函数指针。
pfn_ae_run	运行 AE 的回调函数指针。
pfn_ae_ctrl	控制 AE 内部状态的回调函数指针。
pfn_ae_exit	销毁 AE 的回调函数指针。

pfn_ae_run 函数的参数解释在 2.3 节中做了详细说明。

```
xmedia_s32 (*pfn_ae_ctrl)(xmedia_u32 channel, xmedia_u32 cmd, xmedia_void *value);
```

表15-2 pfn_ae_ctrl 函数的 cmd、vlaue 参数解释

成员名称	描述
XMEDIA_ISP_AAA_CMD_AE_INIT_INFO_GET	获取 ae 初始化使用的配置参数， value 为

	xmedia_isp_ae_result 类型。
XMEDIA_ISP_AAA_CMD_AE_FPS_BASE_SET	设置 sensor 帧率，value 为 xmedia_float 类型。
XMEDIA_ISP_AAA_CMD_AE_BLC_SET	设置 blc 到 ae 算法库中，value 为 xmedia_u16 * 4。
XMEDIA_ISP_AAA_CMD_PROC_WRITE	设置打印 proc 信息参数，value 为空，没有参数。

 说明

调用 xmedia_isp_init 时将调用 pfn_ae_init 回调函数，以初始化 AE 算法库。

当 FE 帧起始中断触发 isp_run 线程的时候将调用 pfn_ae_run 回调函数，以运行 AE 算法库，计算得到 sensor 的曝光时间和增益、ISP 的数字增益。

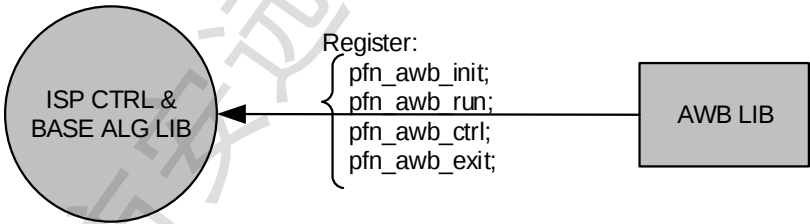
pfn_ae_ctrl 回调函数的目的是改变算法库内部状态。运行时 Firmware 会隐式调用 pfn_ae_ctrl 回调函数，通知 AE 算法库切换 WDR 和线性模式、设置 FPS。

调用 xmedia_isp_exit 时将调用 pfn_ae_exit 回调函数，以销毁 AE 算法库。

15.3 AWB 算法注册 ISP 库

AWB 算法注册 ISP 库调用 xmedia_isp_register_awb_lib，如图 15-2 所示：

图15-2 AWB 算法向 ISP 库注册的回调函数



XMEDIA_AWB 算法实现了一个 xmedia_awb_register 的注册函数，在这个函数中调用 ISP 提供的 xmedia_isp_register_awb_lib 回调接口，用户调用注册函数以实现向 ISP 注册 AWB 算法，示例和 AE 算法库注册类似。

用户需要在自开发定制的 AWB 库中实现以下回调函数：

表15-3 AWB 算法向 ISP 库注册的回调函数

成员名称	描述
pfn_awb_init	初始化 AWB 的回调函数指针。
pfn_awb_run	运行 AWB 的回调函数指针。
pfn_awb_ctrl	控制 AWB 内部状态的回调函数指针。
pfn_awb_exit	销毁 AWB 的回调函数指针。



说明

调用 xmedia_isp_init 时将调用 pfn_awb_init 回调函数，以初始化 AWB 算法库。

当 FE 帧起始中断触发 isp_run 线程的时候将调用 pfn_awb_run 回调函数，以运行 AWB 算法库，计算得到白平衡增益、色彩校正矩阵。

pfn_awb_ctrl 回调函数的目的是改变算法库内部状态。运行时 firmware 会隐式调用 pfn_awb_ctrl 回调函数，通知 AWB 算法库切换 WDR 和线性模式、设置 ISO 和曝光时间。设置 ISO 的目的是为了实现 ISO 与饱和度的联动，增益大时色度噪声也会比较大，所以需要调节饱和度。设置曝光时间是为了 awb 算法判断环境照度，优化场景效果。

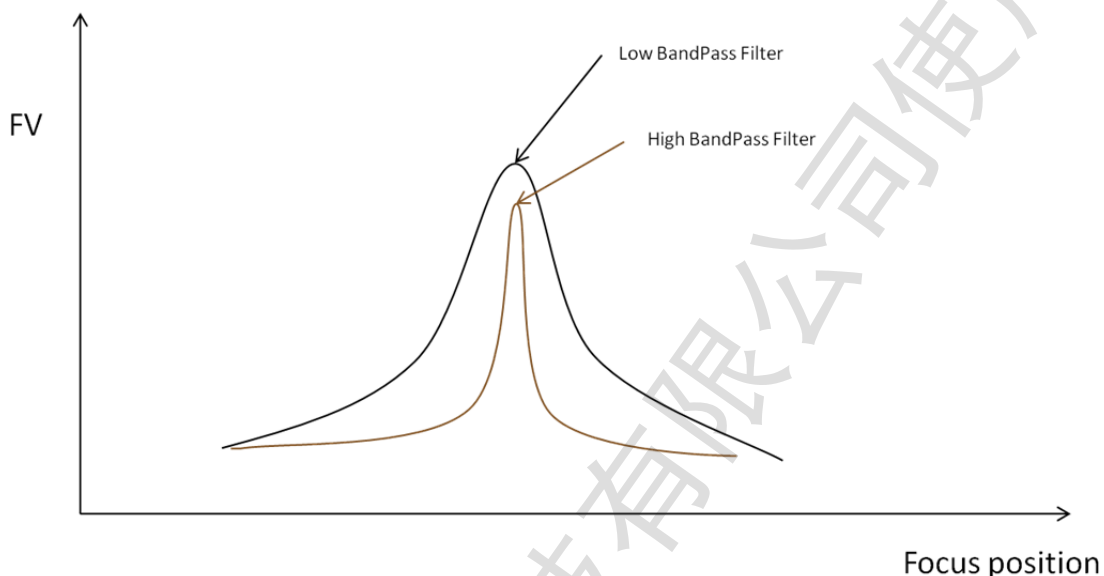
调用 xmedia_isp_exit 时将调用 pfn_awb_exit 回调函数，以销毁 AWB 算法库。

15.4 AF 统计信息使用说明

15.4.1 概述

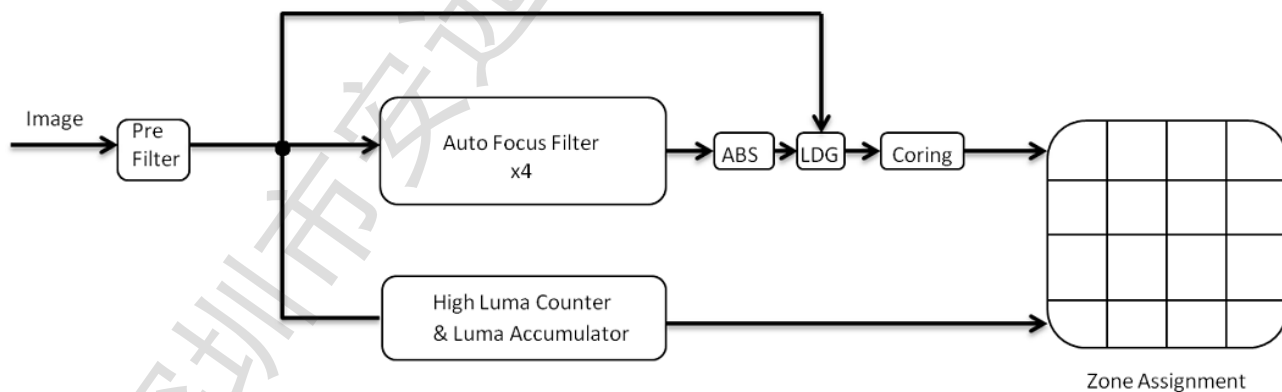
被动式自动对焦一般是通过分析图像特征得出图像清晰度值 FV(Focus Value)，通过驱动对焦马达调节焦点到最佳位置，如图 15-3 所示，图像越清晰 FV 值越大，Peak 点对应聚焦清晰点。获取图像清晰度算法有多种，如灰度梯度法，高频分量法等，目前采用高频分量法来计算 FV，即图像越清晰的时候高频部分幅值越大，将图像通过高通滤波器便可以得到高频分量，一共提供两个 FV 值，以及 Y 和高亮计数器 HICnt。只提供 BE (Behind End) 的 AF 统计信息获取模块。

图15-3 FV 曲线



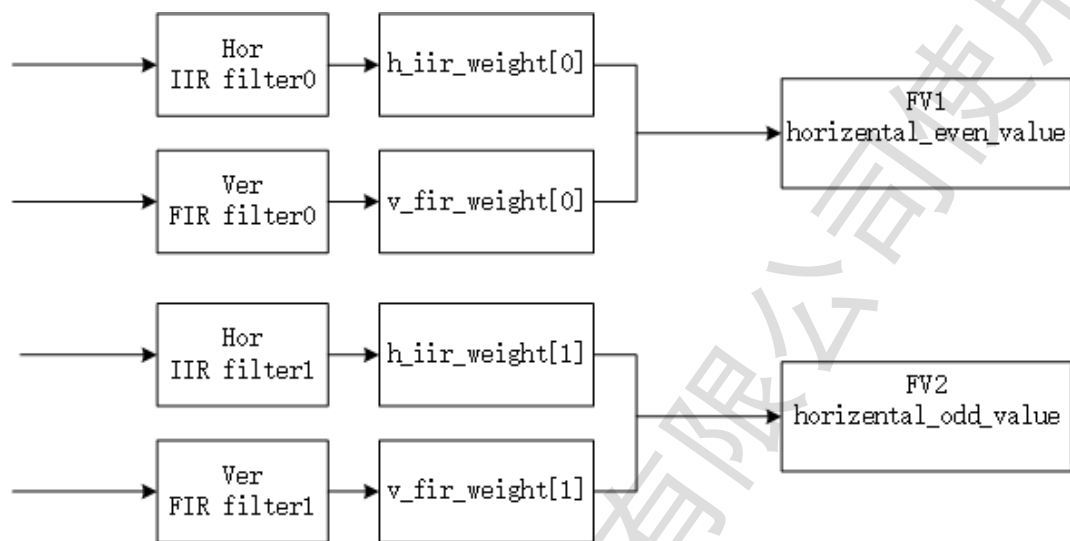
提供 Bayer 域或者 YUV 域进行统计，配置 API 中的 `xmedia_isp_af_switch` 即可以进行统计位置的调整。WDR 模式下，非宽动态场景的 Bayer 域（即 RAW 域）统计信息无明显变化趋势，推荐使用 YUV 域统计信息。首先对图像进行预处理，可在 raw 域配置 `tone_curve` 进行亮度映射，和 `pre_filt` 去除椒盐噪声，在经过水平垂直四组滤波器后，会对输出做 Level Depend Gain 和 Coring，关于这两个模块将在后面的章节详述。模块框图 15-4 所示。

图15-4 AF 统计模块框图



水平垂直滤波器分别采用 IIR 和 FIR 实现如图 15-5 所示。

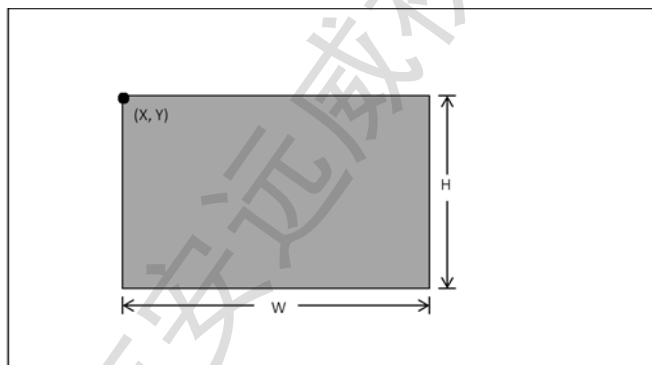
图15-5 AF 4 组滤波器



15.4.2 输入图像的裁剪

AF 模块支持对输入图像的裁剪，如下如图 15-6 用户可以根据实际需求配置 crop 中 X, Y, W, H 四个参数对指定区域进行统计。

图15-6 AF 输入图像裁剪



15.4.3 Bayer 域配置

如果用户将统计位置调整到 Bayer 域则需要配置以下几点。

1. 配置 enPattern 来指定 bayer 数据的 pattern。支持如图 15-7 所示的 4 种 pattern。
2. 配置 AF 模块 gamma 使能，raw 上的线性数据需要做 gamma 处理。gamma 曲线为 xmedia_isp_af_raw_cfg 的 tone_curve，亮度范围为[0,1023]。

图15-7 Bayer 域统计预处理

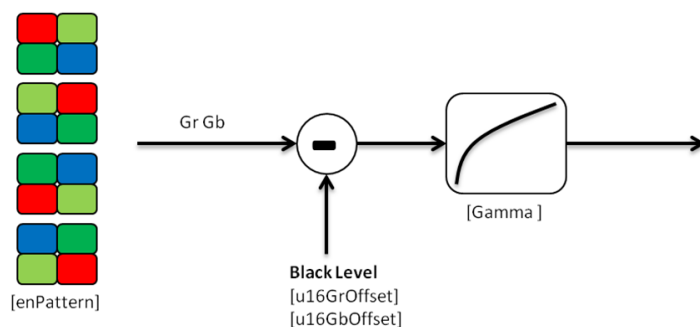
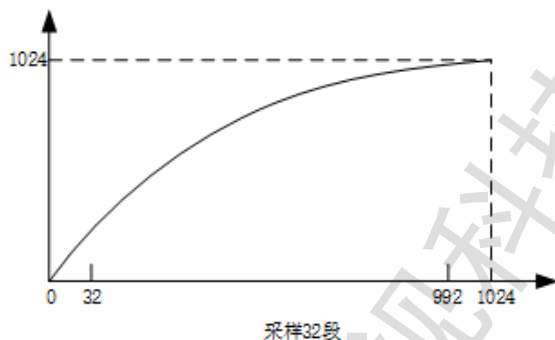


图15-8 关于 Gamma 配置の説明



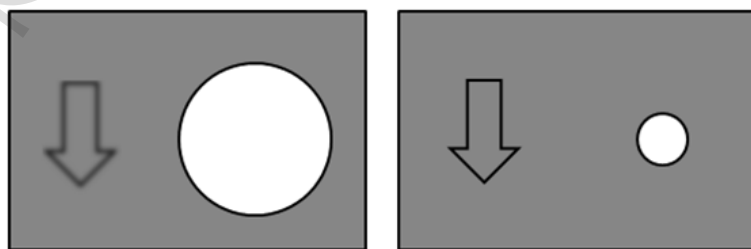
15.4.4 抑制光源对 FV 值的影响

因为 FV 值容易受到 SpotLight 的影响，在聚焦模糊的时候因为光晕扩散，图像中低频分量会增加，会出现图像模糊反而 FV 值变大的现象。为了抑制这一现象，提供以下方法。

- 高亮计数器

如图 15-9，在聚焦模糊的时候因为光晕扩大画面中高亮点的个数增加，清晰的时候高亮点的个数最小，通过配置合适的高亮点判断门限 hlt_thd 即可以辅助实现 SpotLight 场景聚焦。

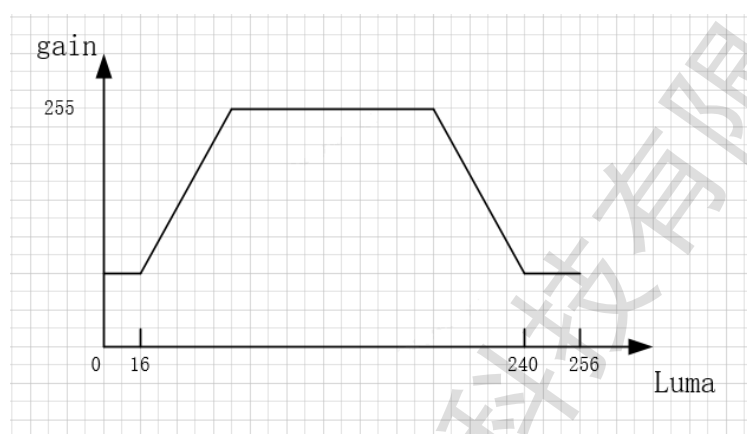
图15-9 高亮统计值



- Level Depend Gain

LDG 的主要原理为通过参考画面像素亮度来衰减滤波器的输出，从而抑制 SpotLight 对 FV 值的影响。如图 15-10 横坐标为像素亮度值，纵坐标为对滤波器输出的 Gain，此处的 Gain 为 8bit 定点小数 Q8。用户可以通过调整 xmedia_isp_af_ld 的 gain_curve 曲线来控制亮暗的衰减系数，255 为原始倍数输出，小于 255 为衰减输出。

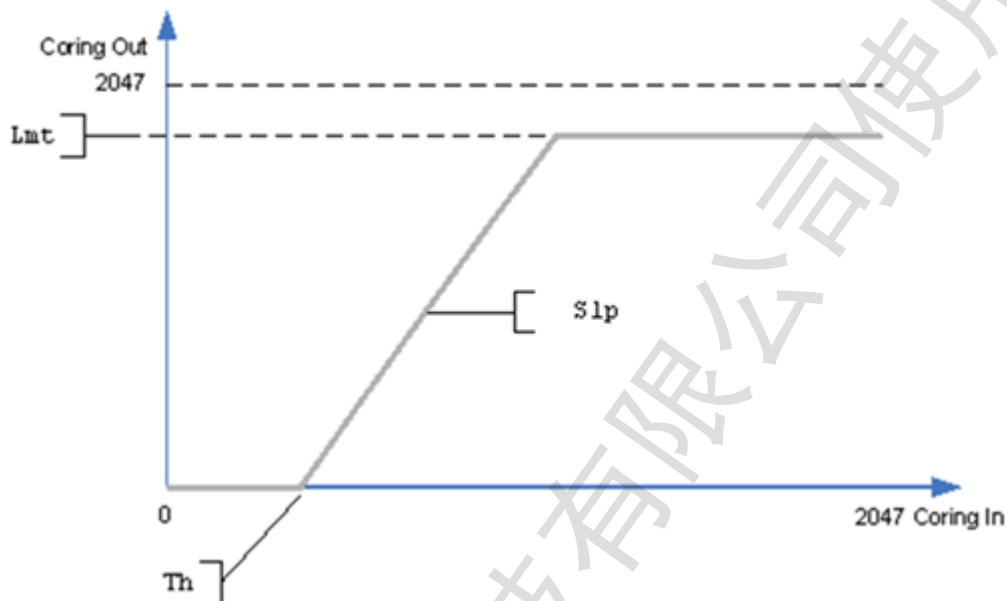
图15-10 Level Depend Gain 曲线



15.4.5 Coring

将滤波器输出通过 Coring 模块可以提高 FV 曲线的抗噪声能力，如图 15-11 所示，设置图中 3 个参数便可以确定 coring 曲线的形状，通过设置合理的参数便可以实现不统计噪声，以及增加有效信号对统计信息的贡献，从而提升统计信息在低照度场景中的性能。

图15-11 Coring 曲线

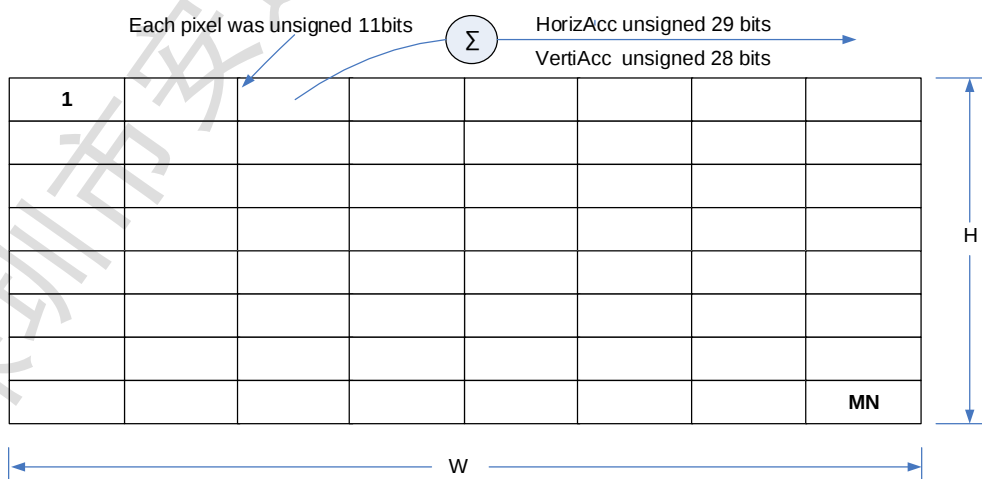


15.4.6 统计模式配置

• Block 大小

对通过滤波器后的图像可以进行分块统计，如图 15-12 所示，块的大小可以设置，最大支持有效个数 18*15 个 blocks，最终块的统计值是对每个 pixel 的累加或者对每行最大值累加。

图15-12 滤波器后的图像



15.4.7 统计信息配置注意事项

表15-4 统计信息配置注意事项

类型	描述
分块大小	有效个数：最大 18 乘以 10
统计模块工作域	RAW/YUV
Bayer 统计模块位置	Raw 在 DGAIN 后、DRC 后
YUV 统计模块位置	CSC 后
Bayer 统计参数是否减去黑电平	已减去
统计输出 YUV 是否减去黑电平	已减去
是否支持 IIR 滤波器 filterShift 补偿	支持
是否支持 IIR 滤波器 narrowband	支持
WDR 模式下特殊配置	YUV 输入时 tone_curve 不生效 RAW 需要配置在 DRC 之后

15.4.8 FV 值计算

一般的推荐最终生成如图 15-3 所示的两条曲线，我们将 low BandPass filter 生成的 FV 命名为 FV1，High BandPass filter 生成的命名为 FV2，FV1 使用在低照度场合，FV2 使用在正常照度场合。对于每一个 block，FV1，FV2 的计算方式为：

$$FV1 = (\alpha * H1 + (32 - \alpha) * V1) / 32;$$

$$FV2 = (\beta * H2 + (32 - \beta) * V2) / 32;$$

H1，H2，V1，V2 分别为水平垂直四组滤波器的统计值，可设置适当的权重对水平和垂直方向的滤波器输出统计值进行混合。

其中， α $32-\alpha$ β $32-\beta$ 对应寄存器 h_iir_weight[0] v_fir_weight[0]
h_iir_weight[1] v_fir_weight[1] 。

最终的 FV 值还需要对每个 block 进行加权，可分别对 block 的 FV1，FV2 采用如下公式计算每帧图像最终的 FV1 FV2 值。

$$FVn = \frac{\sum_{n=1}^{block_n} (FVn * weight_n)}{\sum_{n=1}^{block_n} weight_n}$$

仅限深圳市安远威视科技有限公司使用

16 附录

16.1 注册函数的关系

xmedia_isp_register_ae_lib 和 xmedia_isp_register_awb_lib 这两个接口是 ISP 库提供的钩子函数，用于开发 3A 算法库时实现注册动作。XMEDIA 提供的 3A 算法库 xmedia_ae_register 和 xmedia_awb_register 接口，在实现时调用了相应的钩子函数，所以调用 xmedia_ae_register 能实现 AE 算法库向 ISP 库注册。XMEDIA 的 3A 算法库的注册接口同时完成两件事：3A 库向 ISP 库的注册、sensor 库向 3A 库的注册。

同时，ISP 库也提供了 sensor 库向 ISP 库注册的钩子函数。在 sensor.c 中可以看到对应 sensor 注册函数会调用钩子函数 xmedia_isp_register_sensor 实现 sensor 库向 ISP 库的注册。

XMEDIA 3A 算法库、ISP 库与 sensor 库的注册关系：首先，sensor 库调用 ISP 提供钩子函数向 ISP 注册；其次，ISP 库在 3A 库注册时，将 sensor 注册给 ISP 库的回调函数，通过 copy 的方式推送给 3A 库，实现 sensor 向 3A 库的注册。所以，当用户调用 xmedia_awb_register、xmedia_ae_register 和 xmedia_isp_register_sensor 就完成了 3A 算法库向 ISP firmware 库注册、sensor 库向 3A 算法库和 ISP firmware 库注册。



说明

用户开发 3A 算法库时，请自行实现 3A 库的 xmedia_xx_register 接口，并在该接口实现中调用 xmedia_isp_register_ae_lib 钩子函数，实现 3A 库向 ISP 库的注册功能。

16.2 扩展性的设计考虑

在代码中有 pipe、dev 这样一些概念，这些概念是出于架构扩展性的考虑。

pipe 主要考虑的是支持多个 ISP 单元的情形。无论是多个 ISP 硬件单元，或是一个 ISP 硬件单元分时复用，从软件意义上讲，需要预留出扩展性。

dev 主要考虑的是在多设备驱动中标识一个唯一的设备。

这些概念仅是设计时预留的冗余，如果完全不需要这些概念，可以在开发时去掉这些概念。

A 缩略语

缩写	全称
3A	AE, AWB and AF
AE	Auto Exposure
AF	Auto Focus
API	Application Programming Interface
AWB	Auto White Balance Correction
CAC	Chromatic Aberration Correction
CA	Chroma Adjust
CSC	Color Space Conversion
CCM	Color Correction Matrix
CPU	Central Processing Unit
DIS	Digital Image Stabilization
DPC	Defect Pixel Correction
DRC	Dynamic Range Compression
FPN	Fixed Pattern Noise
FW	Firmware
WDR	Wide Dynamic Range
IMP	Image Processing
ISO	ISO standard 12232:2006. A measure of photographic sensitivity to light
ISP	Image Signal Processor
LUT	Lookup Table
NP	Noise Profile
NR	Noise Reduction

缩写	全称
WB	White Balance
WDR	Wide Dynamic Range
YUV	A method of encoding RGB colorspace in practice equivalent to YCbCr
AFC	Anti False Color
PDAF	Phase Detection Auto Focus
CDAF	Contrast Detection Auto Focus