

密级状态：绝密() 秘密() 内部() 公开(√)

RK IQ XML 参数说明

V5.0

(ISP 部)

文件状态： [√] 正在修改 [] 正式发布	当前版本：	V5.0
	作 者：	欧阳亚凤
	完成日期：	2018-11-20
	审 核：	邓达龙
	完成日期：	2018-12-21

福州瑞芯微电子有限公司

Fuzhou Rockchips Semiconductor Co., Ltd

(版本所有,翻版必究)

版 本 历 史

版本号	作者	修改日期	修改说明	审核	备注
V1.0	欧阳亚凤、池晓芳	2017-05-19	初始版本		rv1108
V2.0	欧阳亚凤	2017-09-17	增加 hal: v1.f.e 对应 IQ 新参数		rv1108
V3.0	欧阳亚凤	2017-09-29	增加无光感 IQ 参数 NLSC_Config		rv1108 Calibdb: v0.1.5
V4.0	欧阳亚凤	2018-09-07	增加 AWB version_11_para 增加 AF 增加 BackLight_Config 增加 Hist_2_hal 增加 NEW_DSP_3DNR_Setting		rv1108 Calibdb: v0.1.9
V5.0	欧阳亚凤	2018-11-09	增加 MFD 增加 UVNR 增加 DPCC 增加 Demosaic_lp_conf 增加 IESHARPEN		Android hal00 后面两个模块参数只有 rk3326 才有

目录

一、文档适用范围.....	5
二、IQ xml 整体结构说明	6
三、模块详细参数说明.....	7
3.1 header 参数说明	7
3.1.1 IQ 基本信息	7
3.1.1.1 code_xml_parse_version.....	7
3.1.1.2 其余 IQ 基本信息.....	8
3.1.2 header 支持配置多个分辨率	8
3.2 sensor 模块参数说明	10
3.2.1 awb 模块参数说明	10
3.2.1.1 globals 参数说明	11
3.2.1.2 illumination 参数说明	17
3.2.2 LSC 参数说明	21
3.2.3 CC 参数说明	22
3.2.4 AF 参数说明.....	23
3.2.5 AEC 参数说明	23
3.2.5.1 AEC 基本参数	24
3.2.5.2 GainRange 、TimeFactor	25
3.2.5.3 ClmTolerance 、Damp	26
3.2.5.4 ExposureSeparate	27
3.2.5.5 AEC_Interval_Adjust_Strategy	28
3.2.5.6 AOE.....	29
3.2.5.6 DON	29
3.2.5.7 NLSC_Config	30
3.2.5.8 BackLight_Config	30
3.2.5.9 Hist_2_hal	31
3.2.6 BLS 参数说明	31
3.2.7 DEGAMMA 参数说明	32
3.2.8 GAMMAOUT/GOC 参数说明	32
3.2.8.1 GAMMAOUT.....	32
3.2.8.2 GOC	33
3.2.9 WDR 参数说明.....	33
3.2.10 CAC 参数说明	35
3.2.11 DPF 参数说明	36
3.2.11.1 DPF	36
3.2.11.2 FilterSetting	37
3.2.11.3 FilterLevelRegConf.....	38
3.2.11.4 Demosaic_th_conf.....	39
3.2.11.5 Demosaic_lp_conf.....	40
3.2.11.6 MFD&UVNR.....	43
3.2.11.7 DSP_3DNR_Setting.....	43
3.2.12 DPCC 参数说明	48

3.2.12.1 dpcc IQ 参数说明	49
3.2.12.3 dpcc 推荐设置	52
3.2.12.4 isp12 修改 dpcc.....	53
3.2.13 CPROC 参数说明.....	56
3.2.14 IESHARPEN 参数说明	57
3.3 system 参数说明	59

dpcc IQ XML V5.0

一、文档适用范围

大部分 RK 平台 IQ xml 参数大部分都是相同的，少部分可能因为平台硬件、代码版本导致有些参数可能不存在或者略有不同。

常用芯片 isp 的区别：

芯片平台	I S P	区别
RK3288/RK3399/RK3368	ISP10	Gamma 17bin
RV1108	ISP11	RKWDR Gamma 34bin
RK3326	ISP12	AE Meas/Hist 9x9 block

常用 Camera 软件说明：

Camera ISP 3A 及 driver 名称	适配系统以及芯片
CameraHal00	RK3399/RK3288/RK3326/RK3368 Android-8.0 及其以前版本
camera_engine_cifisp(3A) cifisp driver	1. RV1108 Linux(Kernel-3.10) 2. RK3288/RK3399 Linux(Kernel-4.4)
camera_engine_rkisp(3A) rkisp driver	1. RK3288/RK3399/RK3326 Linux(Kernel-4.4) 2. RK3399/RK3288/RK3326 Android-9.0 及其后续版本

二、IQ xml 整体结构说明

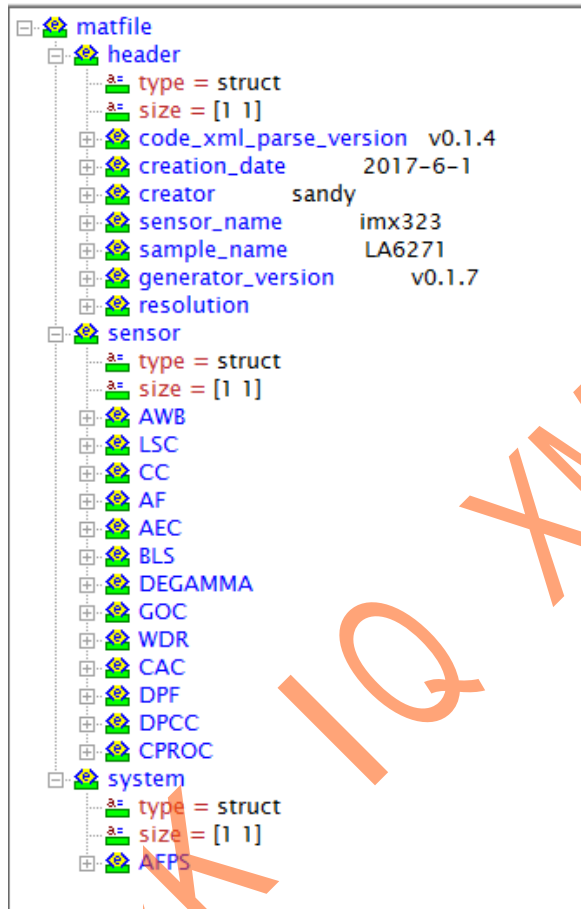
IQ xml 主要由 **header**, **sensor**, **system** 三部分构成, 如下图所示。

Header 主要定义 IQ 创建日期, tuning 负责人, sensor 名称, 镜头名称, IQ xml 版本号和 sensor 支持的分辨率等一些基础信息。

Sensor 是效果参数的主要部分, 包含 AWB, LSC, CC, AF, AEC, BLS, DEGAMMA, GOC, CAC, DPF, DPCC, IESHARPEN 等模块的相应参数。

System 为默认参数, 无需改动。

本文将按照下图参数顺序, 对每个模块的参数进行详细的说明。



Header 新增 `code_xml_parse_version` 代码解析 IQ 文件的版本号, 此版本号必须和运行代码相匹配, 解析才能正确运行, 参数才是匹配的。

三、模块详细参数说明

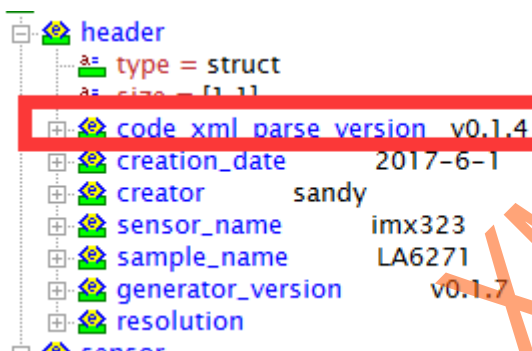
3.1 header 参数说明

Header 主要定义 IQ 创建日期, tuning 负责人, sensor 名称, 镜头名称, IQ xml 版本号和 sensor 支持的分辨率等一些信息。

3.1.1 IQ 基本信息

3.1.1.1 code_xml_parse_version

IQ 相对应代码解析版本号



此版本号为解析 IQ 参数相对应的代码版本号。

如果没有版本号, 或者版本号与代码中不同, 则 camera 会直接退出。

Log 上会有提示, 如下:

```
if(strcmp(meta_data.code_xml_parse_version, CODE_XML_PARSE_VERSION)){
    std::cout
        << "code xml parse version is no match ("
        << meta_data.code_xml_parse_version
        << ") "
        << " != ("
        << CODE_XML_PARSE_VERSION
        << ") "
        << std::endl;
    return (false);
}
```

升级代码同时, 相应 IQ 中的版本号也必须升级, 相应参数要对比进行修改。

代码中对应解析参数版本号为: \CameraHal10_Release\CameraHal\calib_xml\calibdb.cpp 中:

3.1.1.2 其余 IQ 基本信息

```
<header size="[1 1]" type="struct">
  <creation_date index="1" size="[1 10]" type="char">
    2016-08-31
  </creation_date>
  <creator index="1" size="[1 4]" type="char">
    OYYF
  </creator>
  <sensor_name index="1" size="[1 6]" type="char">
    OV4689
  </sensor_name>
  <sample_name index="1" size="[1 9]" type="char">
    LA6111PA
  </sample_name>
  <generator_version index="1" size="[1 6]" type="char">
    v0.4.0
  </generator_version>
```

IQ创建日期

调试人员名字

sensor名称

镜头名称

IQ版本

3.1.2 header 支持配置多个分辨率

RK 在 Sensor 实际效果调试一般会调试两个分辨率，一个作为预览用的小分辨率（一般为 Binning 分辨率），一个作为拍照用的全分辨率（一般为 FULL 分辨率）。车机或者无人机的产品一般只调试一个分辨率。

需要支持更多或更少时，需要将 resolution 字段的定义进行相应修改，并对 sensor 参数中依赖分辨率的其他参数进行修改。

每个分辨率为一个 cell 依次填写在 resolution 字段内。


```

</generator_version>
<resolution index="1" size="[1 2]" type="cell">
  <cell index="1" size="[1 1]" type="struct">
    <name index="1" size="[1 9]" type="char">
      1920x1080
    </name>
    <id index="1" size="[1 10]" type="char">
      0x00000001
    </id>
    <width index="1" size="[1 1]" type="double">
      [ 1920]
    </width>
    <height index="1" size="[1 1]" type="double">
      [ 1080]
    </height>
    <framerate index="1" size="[1 3]" type="cell">
      <cell index="1" size="[1 1]" type="struct">
        <name index="1" size="[1 6]" type="char">
          FPS_15
        </name>
        <fps index="1" size="[1 1]" type="double">
          [ 15.0000]
        </fps>
      </cell>
    </framerate>
  </cell>
</resolution>
</generator_version>
<resolution index="2" size="[1 2]" type="cell">
  <cell index="2" size="[1 1]" type="struct">
    <name index="1" size="[1 9]" type="char">
      2688x1520
    </name>
    <id index="1" size="[1 10]" type="char">
      0x00000002
    </id>
    <width index="1" size="[1 1]" type="double">
      [ 2688]
    </width>
    <height index="1" size="[1 1]" type="double">
      [ 1520]
    </height>
    <framerate index="1" size="[1 1]" type="cell">
      <cell index="1" size="[1 1]" type="struct">
        <name index="1" size="[1 6]" type="char">
          FPS_08
        </name>
        <fps index="1" size="[1 1]" type="double">
          [ 7.5000]
        </fps>
      </cell>
    </framerate>
  </cell>
</resolution>

```

支持多个分辨率

第一个分辨率

分辨率名称由宽x高命名

分辨率id号, 按顺序从1递增

分辨率的宽

分辨率的高

分辨率序列对于帧率命名为FPS_帧率

帧率

第二个分辨率

3.2 sensor 模块参数说明

3.2.1 awb 模块参数说明

目前 awb 参数有两个版本，版本适配情况如下：

AWB 版本	软件适配情况
AWB VERSION_10	camera_engine_rkisp, CameraHal00 camera_engine_cifisp
AWB VERSION_11	camera_engine_cifisp

两者的区别点如下：

AWB VERSION_10：白点条件参数还需经过多次转换，最后和寄存器条件相同。判断光源采用二维高斯分布方式来判断，和白平衡范围图上采用不同坐标，比较不直观。

AWB VERSION_11：白点条件参数直接使用寄存器相对应的条件，更为直观，方便调试。判断光源方式采用欧式距离方式判断，比较直观判断光源。

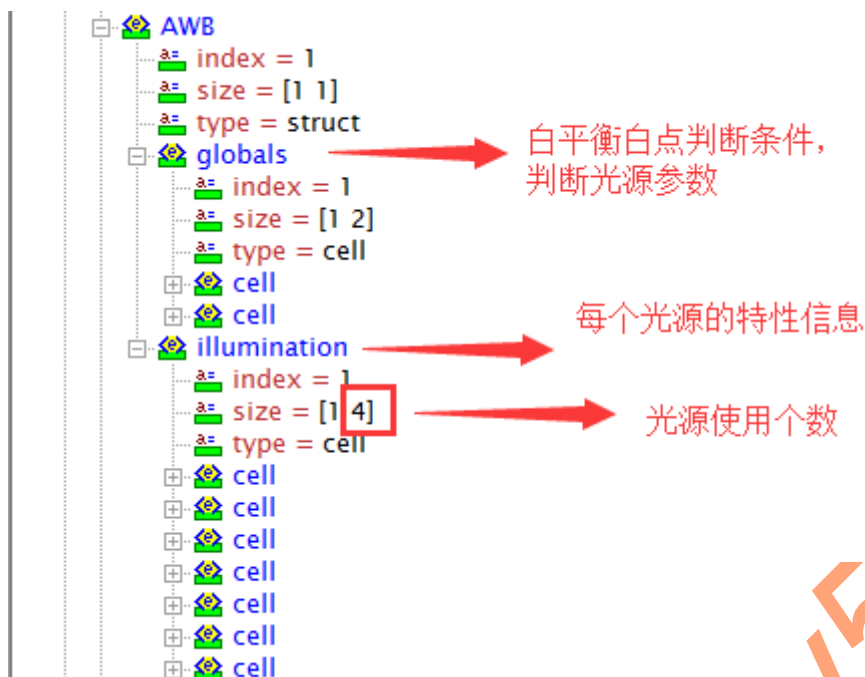
Awb 模块包含 **globals** 和 **illumination** 两部分参数。

globals 主要为白平衡白点条件参数和判断光源相关参数。

一般有几个分辨率就配置几个 **cell**，每个 **cell** 的参数基本相同，除了分辨率不同而已。如上所说，rk 调试一般选择两个分辨率，所以 **cell** 一般为 2 个。

illumination 为调试时候选择的几种标准光源的特性信息，包括白平衡，**ccm**，**lsc** 插值和 **cc** 插值相关信息，以及光源特性参数。

Size 必须和实际调试光源个数相匹配。**Xml** 解析会按照填写的个数去解析，多余的不解析。rk 调试一般只选择 **A**、**CWF**、**TL84**、**D65** 四个标准光源而已，所以填写 4。

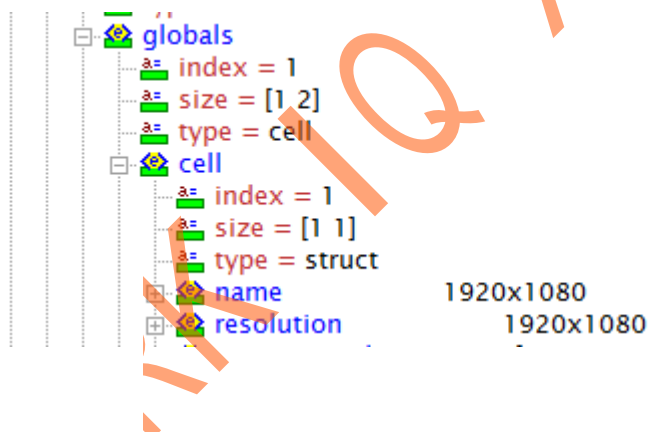


3.2.1.1 globals 参数说明

3.2.1.1.1 分辨率信息参数

Name 命名为 width x height

Resolution 命名也为 width x height



3.2.1.1.2 光源判断参数

1: AWB VERSION_10 光源判断参数

在做白平衡 tuning 是会产生此参数，将其填入如下位置。

SVDMeanValue	[0.3405073885950044 0.4
PCAMatrix	[-0.6787904814407830 -0.053
CenterLine	[-0.7251382973142543 -0.681

SVDMeanValue 与 PCAMatrix 均为 AWB 模块 tuning 后得到的结果。

SVDMeanValue: 为多个光源下的白点对应的 R,G,B 三通道的均值。

PCAMatrix: 为多个光源下白点的 R,G,B 三通道均值确定的特征空间。

这边引入 PCA（主成分分析）的目的是让各个光源在 PCAMatrix 特征空间下的投影的方差最大，也就是每个光源在该特征空间的投影是可区分的，那么对于任意一个场景，用该场景的下白点的 R,G,B 三通道均值可以得到其在特征子空间上的投影，进而识别该场景是何种光源。在做白平衡 tuning 是会产生此参数，将其填入如下位置。

2: AWB VERSION_11 光源判断参数

不再有上面 AWB VERSION_10 对应参数。

直接通过统计的白平衡 rgain bgain 坐标,计算和标准光源距离，最小距离为此情景对应光源。

3.2.1.1.3 白平衡范围大小参数

白平衡会对整个白平衡区域做范围大小限制，tuning 过程中会产生此参数，请将其填入如下位置。

+	afRg2	[1.13595f 1.29387f 1.40866f 1.51355f
+	afMaxDist2	[0.15888f 0.14051f 0.14973f 0.14051f
+	afRg1	[1.15000f 1.30834f 1.42014f 1.51489f
+	afMaxDist1	[0.04458f 0.08683f 0.11337f 0.12014f
+	afGlobalFade2	[0.80000f 0.91376f 1.02751f 1.02751f
+	afGlobalGainDistance2	[0.56214f 0.50946f 0.50946f 0.56214f
+	afGlobalFade1	[0.80000f 0.91376f 1.02751f 1.02751f
+	afGlobalGainDistance1	[0.03786f 0.09054f 0.09054f 0.03786f

上面的这几个参数均为 tuning 后得到的，与白平衡 tuning 时的绘制的白点区域相关，详细描述如下图：

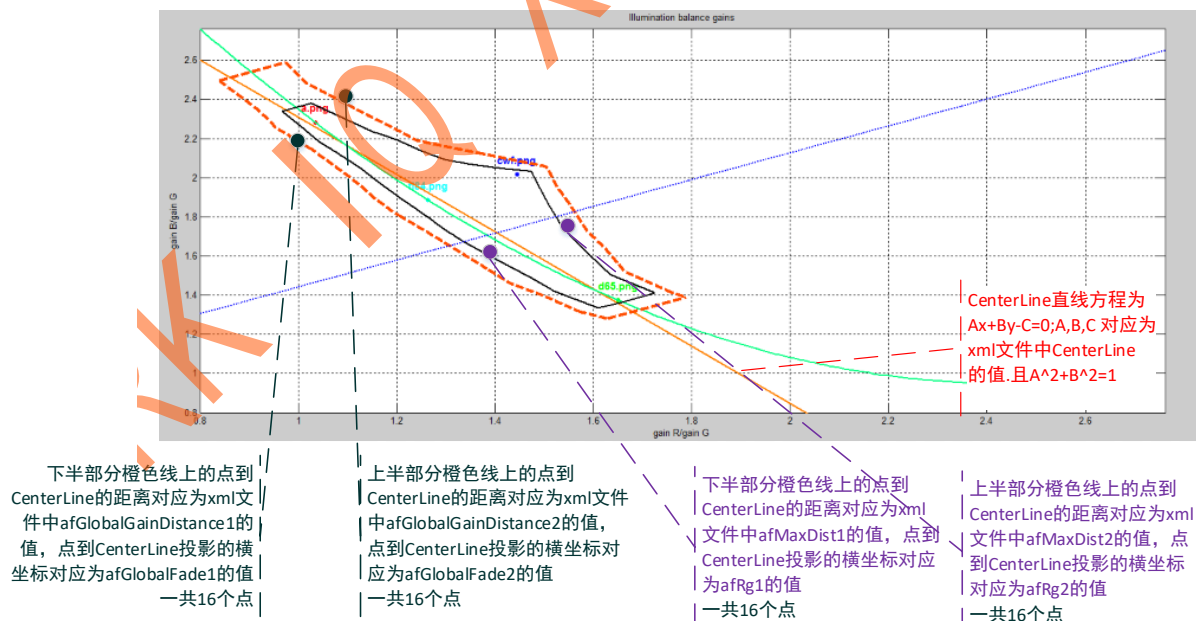


图-1 白点区域曲线

当得到该场景的 (gainR/gainG, gainB/gainG) 后并不是直接运用于白平衡校正,而是会进行一定的调整。首先会判断该点是否超出了上图中橙色虚线的范围,接着用 IIR 滤波器将当前的该场景的 (gainR/gainG, gainB/gainG) 与前一帧的 (gainR/gainG, gainB/gainG) 关联起来 (具体的实现见 AwbIIRDampCoef 参数的描述部分), 最后用上图中的黑色虚线将 (gainR/gainG, gainB/gainG) 限制在黑色虚线的范围内




















3.2.1.1.4 曝光和白点范围条件

1: AWB VERSION_10 白点条件参数说明

ISP 内部硬件会根据白点条件统计出所有符合条件的白点的 RGB 三通道均值。

软件上 AWB 会根据三通道的统计均值去判断当前为何种光源，该采用哪个光源的参数。

所以白点条件是影响白平衡最终结果以及选择光源的非常重要的影响因子。

	K_Factor	[4.5676]
	afFade2	[1.2 1.55 1.8 1.95 2.1 2.35]
	afCbMinRegionMax	[115 110 105 100 96 93]
	afCrMinRegionMax	[90 95 105 113 120 125]
	afMaxCSumRegionMax	[20 16 14 12 12 10]
	afCbMinRegionMin	[128 126 124 122 120 118]
	afCrMinRegionMin	[90 95 105 116 123 126]
	afMaxCSumRegionMin	[12 10 9 9 9 9]
	afMinCRegionMax	[20 17 15 15 14 13]
	afMaxYRegionMax	[230 230 230 230 230 230]
	afMinYMaxGRegionMax	[30 30 30 30 30 30]
	afMinCRegionMin	[16 14 13 12 12 10]
	afMaxYRegionMin	[230 230 230 230 230 230]
	afMinYMaxGRegionMin	[30 30 30 30 30 30]
	afRefCb	[128 128 128 128 128 128]
	afRefCr	[128 128 128 128 128 128]
	RegionSize	[1]
	RegionSizeInc	[0.3]
	RegionSizeDec	[0.02]

K_Factor:

此参数与曝光相关，当环境亮度达到一定程度，会默认判断为室外光源，屏蔽室内光源影响。
计算光源为室外光源的概率：

$\text{ExpPriorOut} = \text{func}(\text{fGExp})$,

$\text{fGExp} = (\text{SensorGain} * \text{IntegrationTime} * \text{K_Factor})$;

曝光参数越大为室外光源的概率就越小。

若 $\text{ExpPriorOut} \geq 1$, 则判定为室外光源，

若 $\text{ExpPriorOut} \leq 0.5$, 则判定为室内光源。

若 $0.5 < \text{ExpPriorOut} < 1$, 则该光源即可能是室外光源也可能是室内光源。

光圈大的 sensor 在 A 光下需要的曝光比较小，容易导致 $\text{ExpPriorOut} > 1$, 被判定为室外光源。
此时可以增大 K_Factor，减小 ExpPriorOut ，使得 A 光不会被判断为室外光源。

f_CbMin_regionMax, f_CbMin_regionMin,

f_CrMin_regionMax, f_CrMin_regionMin,

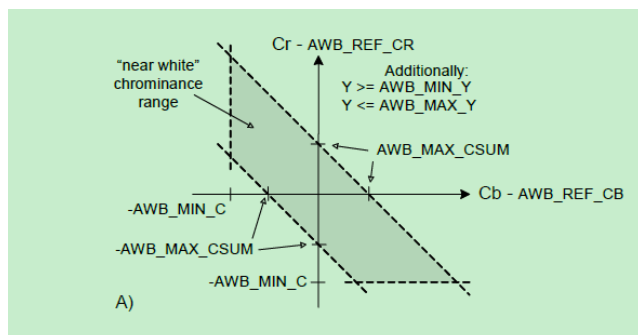
f_MaxCSum_regionMax, f_MaxCSum_regionMin

上面 6 个参数矩阵中不同位置的值与 afFade2 矩阵中相应位置的值一一对应。

这些参数并不是 tuning 后得到的，需要参考白平衡范围坐标进行微调。

afFade2 中的值为多个 gainR/gainG 值下在 CenterLine 投影的横坐标。

用这些参数确定不同 gain 下白点的边界条件的 AWB_REF_CR (RefCr_MaxR), AWB_REF_CB (RefCb_MaxB), AWB_MIN_C (MinC), AWB_MAX_CSUM (MaxCSum), AWB_MAX_Y (fMaxY), 图示变量名 (程序中的变量名)



由 RgProj (当前场景的 gainR/gainG 在 CenterLine 投影的横坐标) 插值得到该场景下对应的 f_CbMin_regionMax, f_CbMin_regionMin, f_CrMin_regionMax, f_CrMin_regionMin, f_MaxCSum_regionMax, f_MaxCSum_regionMin 值, 按下面的式子计算下一帧的白点的边界条件:

$$f_CbMin = \text{RegionSize} * f_CbMin_regionMax + (1.0f - \text{RegionSize}) * f_CbMin_regionMin$$

$$f_CrMin = \text{RegionSize} * f_CrMin_regionMax + (1.0f - \text{RegionSize}) * f_CrMin_regionMin$$

$$f_MaxCSum = \text{RegionSize} * f_MaxCSum_regionMax + (1.0f - \text{RegionSize}) * f_MaxCSum_regionMin$$

$$\text{MeasWdw.RefCr_MaxR} = 0.5 * (f_CrMin - f_CbMin) + 128$$

$$\text{MeasWdw.RefCb_MaxB} = 0.5 * (f_CbMin - f_CrMin) + 128$$

$$\text{MeasWdw.MinC} = -0.5 * (f_CbMin + f_CrMin) + 128$$

$$\text{MeasWdw.MaxCSum} = f_MaxCSum$$

其中 RegionSize 根据当前帧的白点数量进行调整,

如果白点数量小于白点数量的最小值

$$\text{RegionSize} = \text{MIN}(1.0f, (\text{RegionSize} + \text{RegionSizeInc}))$$

如果白点数量大于白点数量的最大值且白点在外部的白点边界线外

$$\text{RegionSize} = \text{MAX}(0.0f, (\text{RegionSize} - \text{RegionSizeDec}))$$

概况的说: 白平衡白点条件随着 afFade2 参数的变化进行变化。

afCbMinRegionMax 和 **afCbMinRegionMin**: 影响白点的 refcb 分量条件。

afCrMinRegionMax 和 **afCrMinRegionMin**: 影响白点的 refcr 分量条件。

afMaxCSumRegionMax 和 **afMaxCSumRegionMin**: 影响和分量条件。

afMinCRegionMax 和 **afMinCRegionMin**: 影响单个分量条件。

afMaxYRegionMax 和 **afMaxYRegionMin**: 影响最大 Y 限制。

afMinYMaxGRegionMax 和 **afMinYMaxGRegionMin**: 影响最小 Y 限制。

afRefCb 和 **afRefCr**: 影响白点的 refcb 和 refcr 分量基准, 一般为 128。

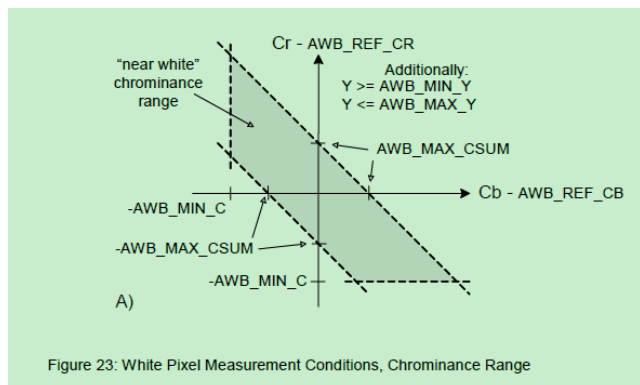
此条件需根据 sensor 调试的 awb 范围坐标由 IQ 调试人员进行相应改变, 并非 tuning 会自动产生的参数。

RegionSize、**RegionSizeInc**、**RegionSizeDec**:

此三个参数受白点检测出数量影响, 从而影响上面白平衡条件。

2: AWB VERSION_11 白点条件参数说明

为了使白点条件调节更为直观有效，方便调试，直接使用下图边界参数作为调试参数。



下面是对 VERSION_11 版本白点条件描述。

afFade2	[1.2 1.55 1.8 2.05 2.3 2.5]
afmaxCSum_br	[20 16 14 12 12 10]
afmaxCSum_sr	[12 10 9 9 9 9]
afminC_br	[20 17 15 15 14 13]
afMaxY_br	[230 230 230 230 230 230]
afMinY_br	[30 30 30 30 30 30]
afminC_sr	[16 14 13 12 12 10]
afMaxY_sr	[230 230 230 230 230 230]
afMinY_sr	[30 30 30 30 30 30]
afRefCb	[140.5 134 128 126.5 120.5 113]
afRefCr	[115.5 122 128 129.5 135.5 143]

afFade2: afFade2 中的值为 awb 白平衡范围图上多个 gainR/gainG 值下在 CenterLine 投影的横坐标。用来控制不同色温下使用不同的白点条件。

afmaxCSum_br, afmaxCSum_sr:

白点边界图上 AWB_MAX_CSUM, br sr 后缀为, 白点数量多和少时候对应插值边界。

afminC_br, afminC_sr:

白点边界图上 AWB_MIN_C, br sr 后缀为, 白点数量多和少时候对应插值边界。

afMaxY_br, afMaxY_sr:

白点边界图上 AWB_MAX_Y, br sr 后缀为, 白点数量多和少时候对应插值边界。













afMinY_br, afMinY_sr:

白点边界图上 AWB_MIN_Y, br sr 后缀为, 白点数量多和少时候对应插值边界。

afRefCb: 白点边界图上 AWB_REF_CB。

afRefCr: 白点边界图上 AWB_REF_CR。

3.2.1.1.5 白平衡范围特殊限制

	fRgProjIndoorMin	[1.15]
	fRgProjMax	[2.4663]
	fRgProjMaxSky	[2.55]
	fRgProjOutdoorMin	[1.9]
	fRgProjYellowLimitEnable	[0]
	fRgProjALimit	[5.0]
	fRgProjAWeight	[0.0]
	fRgProjYellowLimit	[1.4]
	fRgProjIllToCwfEnable	[0]
	fRgProjIllToCwf	[-0.15]
	fRgProjIllToCwfWeight	[0.0]
	awb_clip_outdoor	D65

fRgProjIndoorMin: 最小可设定白平衡范围。

为图-1 中黑色虚线左端与 CenterLine 的交点,为室内光源下 RgProj 的最小值。

fRgProjMax: 最大可设定白平衡范围。

为图-1 中黑色虚线右端与 CenterLine 的交点, 为 RgProj 的最大值。

fRgProjMaxSky: 最大可检测白平衡范围。

为图-1 中橙色虚线右端与 CenterLine 的交点,代表的是白天蓝天的 gainR/gainG 在 CenterLine 投影的横坐标, 当 gainR/gainG 在 CenterLine 上投影的横坐标大于该值时判定改点超出白点的区域。

fRgProjOutdoorMin: 室内外分界白平衡界限。

为图-1 中斜的蓝色虚线与 CenterLine 的交点, 为室外光源下 RgProj 的最小值。

上面四个参数均为 tuning 过程中, 会自动产生参数。

(1) 当判定该场景的光源为室内光源时, 调整后的 gainR/gainG 在 CenterLine 上投影的横坐标均大于等于 fRgProjIndoorMin;

(2) 当判定该场景的光源为室外光源时, 调整后的 gainR/gainG 在 CenterLine 上投影的横坐标均大于等于 fRgProjOutdoorMin;

(3) 当该场景的光源可能为室外光源也可能为室内光源时, 调整后的 gainR/gainG 在 CenterLine 上投影的横坐标均大于等于 fRgProjOutdoorMin 与 fRgProjIndoorMin 加权值的值。当前场景为室内或室外光源的判定见下面的补充部分的描述。

(4) 所有场景下, 调整后的 gainR/gainG 在 CenterLine 上投影的横坐标均小于等于 RgProjMax。

下面 7 个参数均为特殊处理参数, 一般是保持关闭状态, 仅在特殊状况下会使用。通过 fRgProjYellowLimitEnable 和 fRgProjIllToCwfEnable 设置为 0 关闭, 设置 1 为打开。

fRgProjYellowLimitEnable、fRgProjALimit、fRgProjAWeight、fRgProjYellowLimit:

改变对黄色物体白平衡微调。

fRgProjIllToCwfEnable、fRgProjIllToCwf、fRgProjIllToCwfWeight:

是对 cwf 和 d65 光源判断一个调节。

此参数需要人工对 sensor 的白平衡信息进行调节, 非自动化产生参数。

awb_clip_outdoor 此参数无需改变。

3.2.1.1.6 IIR 参数

IIR 主要包括两部分参数：一部分是 DAMP 参数，一部分是 ExpPrior 参数。
此部分参数一般无需改变。

DAMP: 通过当前帧与前一帧白平衡值进行加权平均，影响白平衡调节的快慢速度。

DampCoefAdd: damp 系数加加时每次加的值。

DampCoefSub: damp 系数减减时每次减的值。

DampFilterThreshold: 满足 damp 系数加加或减减条件。

DampingCoefMin: damp 系数最小值。

DampingCoefMax: damp 系数最大值。

DampingCoefInit: damp 系数初始值。

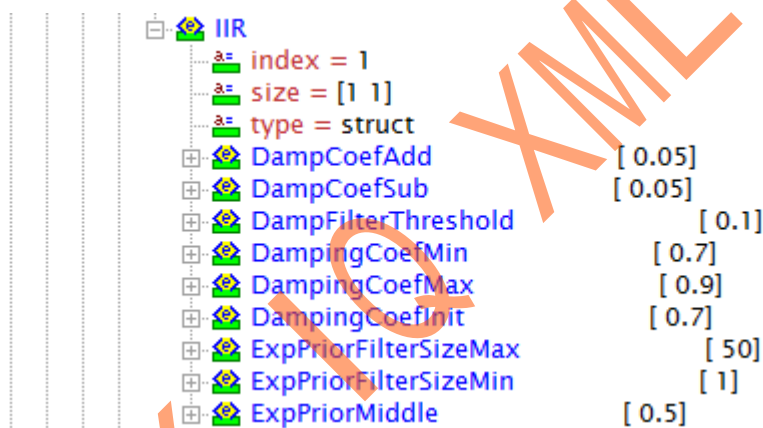
ExpPrior: 基本保持不变，会记录 50 帧的 ExpPriorIn 参数，对其求均值 mean。

通过判断 mean - 当前 ExpPriorIn 是否超过 DampFilterThreshold，来决定 DAMP 系数增减，影响白平衡调节的快慢速度。

ExpPriorFilterSizeMax: ExpPrior 最大记录帧数，代码已定最大值为 50，此参数不可大于 50。

ExpPriorFilterSizeMin: ExpPrior 最小记录帧数。

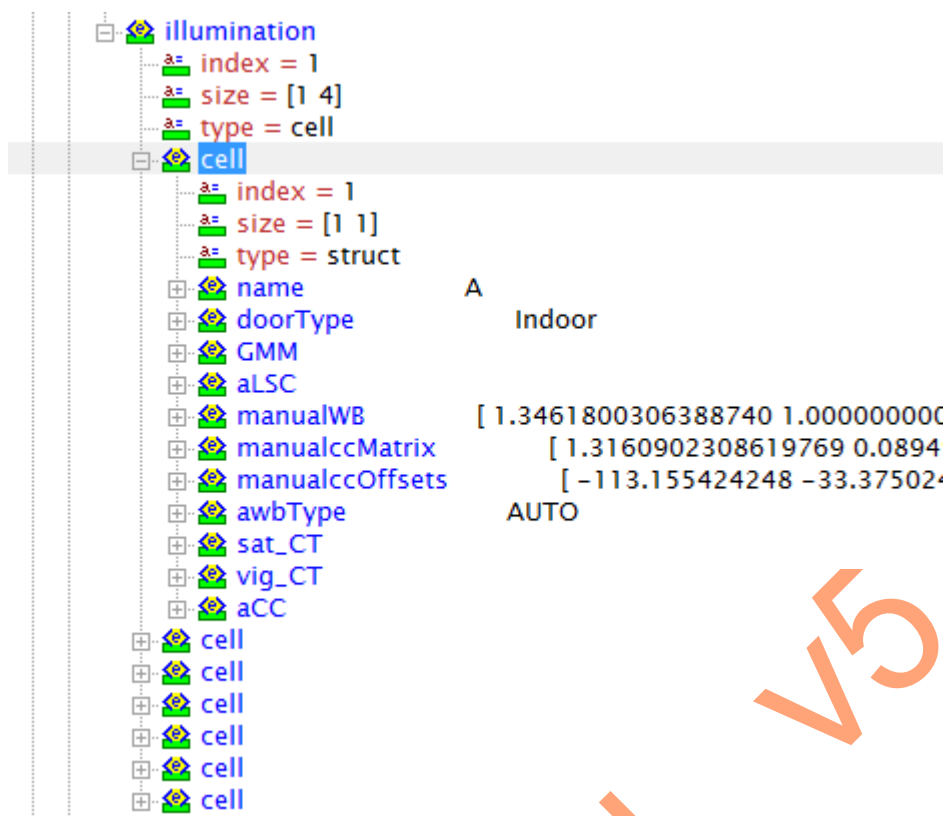
ExpPriorMiddle:



IIR	
a= index = 1	
a= size = [1 1]	
a= type = struct	
DampCoefAdd	[0.05]
DampCoefSub	[0.05]
DampFilterThreshold	[0.1]
DampingCoefMin	[0.7]
DampingCoefMax	[0.9]
DampingCoefInit	[0.7]
ExpPriorFilterSizeMax	[50]
ExpPriorFilterSizeMin	[1]
ExpPriorMiddle	[0.5]

3.2.1.2 illumination 参数说明

每个 cell 保存相应光源的特性信息，这些特性信息主要为白平衡，色彩校正矩阵，lsc 和 cc 插值矩阵。



Name: 光源名称

doorType: 区分室内室外光源，室内光源为 Indoor，室外光源 Outdoor。

GMM: 高斯概率参数，决定落在该光源内的概率。

alsc: 该光源支持几个分辨率和几种补偿百分比的 lsc 调试参数。

manualWB: 该光源白平衡数值。

manualccMatrix: 该光源色彩校正矩阵。

manualccOffsets: 该光源色彩校正矩阵的 offset 参数。

awbType: 默认都是 AUTO，自动白平衡。

sat_CT: 根据不同曝光的 gain 值，可对选择不同曝光进行不同饱和度设定策略。

vig_CT: 根据不同曝光的 gain 值，可对选择不同曝光进行不同 lsc 补偿程度设定策略。

aCC: 色彩校正矩阵支持的几种饱和度的校正信息。

3.2.1.2.1 invCovMatrix, GaussianScalingFactor, tau

```
<cell index="1" type="struct" size="[1 1]">
  <name index="1" type="char" size="[1 1]">
    A
  </name>
  <doorType index="1" type="char" size="[1 6]">
    Indoor
  </doorType>
  <GMM index="1" type="struct" size="[1 1]">
    <invCovMatrix index="1" type="double" size="[1 4]">
      [3353.7255084580888000 3671.2040650236199000
       3671.2040650236199000 6405.5918746894431000]
    </invCovMatrix>
    <GaussianScalingFactor index="1" type="double" size="[1 1]">
      [450.2948047893071400]
    </GaussianScalingFactor>
    <tau index="1" type="double" size="[1 2]">
      [0.85 0.9]
    </tau>
    <GaussianMeanValue index="1" type="double" size="[1 2]">
      [-0.0691687425107657 0.0113525496137357]
    </GaussianMeanValue>
  </GMM>
</cell>
```

这部分参数用于计算当前光源为某种光源的概率（后验概率），均由 tuning 后得到的。上图给出的是 A 光源下的相关参数。

3.2.1.2.2. manualWB, manualccMatrix, manualccOffset, sat_CT, gains

```
<manualWB index="1" type="double" size="[1 4]">
  [1.4441266249921583 1.0000000000000000 1.0000000000000000 2.0159169373382961]
</manualWB>
<manualccMatrix index="1" type="double" size="[3 3]">
  [ 1.9080451178641047 -1.1228523125418279 0.2148071946777233
    -0.3451041571060539 1.7658749663805153 -0.4207708092744613
    -0.1107384323926074 -0.8220006914443223 1.9327391238369298]
</manualccMatrix>
<manualccOffsets index="1" type="double" size="[1 3]">
  [-50 -50 -50]
</manualccOffsets>
<awbType index="1" type="char" size="[1 4]">
  AUTO
</awbType>
<sat_CT index="1" type="struct" size="[1 1]">
  <gains index="1" type="double" size="[1 4]">
    [1 2 4 8]
  </gains>
  <sat index="1" type="double" size="[1 4]">
    [100 100 100 100]
  </sat>
</sat_CT>
<vig_CT index="1" type="struct" size="[1 1]">
  <gains index="1" type="double" size="[1 4]">
    [1 2 4 8]
  </gains>
  <vig index="1" type="double" size="[1 4]">
    [100 100 100 100]
  </vig>
</vig_CT>
```

上面这些参数主要用于计算 ACC 部分的色彩校正参数，下面将逐一介绍。

manualWB:

为各个光源下 tuning 后得到的白平衡校正参数，对应于 R,Gr,Gb,B 通道的调节参数。

manualccMatrix 与 manualccOffset:

为各个光源下选择 100%或者 74%的饱和度进行 tuning 后得到的参数。

sat_CT 表示饱和度与 gains 一一对应。

当确定了当前场景所落在的区域，获取 ACC 对应的 CCM 过程如下：

1: 若为区域 A，CCM 只由主导光源确定

1.1: 已知当前帧的 gain 利用 sat 和 gains 进行插值得到对应的 saturation，即通过 gain 来确定当前场景对应的 saturation 值；

1.2: 根据 saturation 得到相应的 CCM 矩阵，一般我们会 tuning 每个光源下 saturation=74%和 saturation=100%的两个 CCM 相关的矩阵。利用主导光源的这两个矩阵，用插值的方法得到主导光源下的 saturation 对应的 CCM 矩阵。

1.3: 根据 saturation 得到相应的 CCMoffset 矩阵，与 1.2 类似。

2: 若为区域 B 或 C，CCM 由 tuning 时用到的所有光源共同决定，当为区域 B 时主导光源的贡献更多。

首先，按下列步骤计算出每一个光源 i 对应的 CCM_i 矩阵，offset_i 矩阵

2.1: 已知当前帧的 gain 利用 sat 和 gains 进行插值得到对应的 saturation，即通过 gain 来确定当前场景对应的 saturation 值；

2.2: 根据 saturation 得到相应的 CCM_i 矩阵，一般我们会 tuning 每个光源下 saturation=74%和 saturation=100%的两个 CCM 相关的矩阵。利用主导光源的这两个矩阵，用插值的方法得到该光源下的 saturation 对应的 CCM_i 矩阵。

2.3: 根据 saturation 得到相应的 CCMoffset_i 矩阵，与 2.2 类似。

2.4: 由权值 WeightTrans_i 对所有光源的矩阵进行加权和得到最终未 damp 的矩阵 CCM 及 CCMoffset。其中 WeightTrans_i 的计算由各个光源的后验概率 $p(Illu_i | fPca)$ 及 tau1,tau2 决定。

2.5: 对矩阵进行 damp，用前一帧的矩阵和当前帧的矩阵进行加权和确定最终的矩阵，权值为 AwbIIRDampCoef（计算白平衡的校正参数也用到这个参数）。

3.2.1.2.3 vig_CT

```
<vig_CT index="1" type="struct" size="[1 1]">
  <gains index="1" type="double" size="[1 4]">
    [1 2 4 8]
  </gains>
  <vig index="1" type="double" size="[1 4]">
    [100 100 100 100]
  </vig>
</vig_CT>
```

vig_CT 表示图像四周的亮度与图像中心的亮度的比值，vig_CT 与 gains 一一对应，及对应的 gain 下校正后的图像图像四周的亮度与图像中心的亮度的比值为 vig_CT 中对应的值。

当确定了当前场景对应的曝光 gain 后，获取 ALSC 对应的校正参数过程如下：

(a) 已知该场景对应的 gain，利用 vig_CT 和 gains 进行插值得到对应的 vig，即通过 gain 来确定当前场景对应的 vig 值；

(b) 根据 vig 得到相应的 ALSC 校正参数矩阵，一般我们会 tuning 每个光源下 vig=70%和 vig=100%的两个 LSC 参数。利用主导光源的这两个参数，用插值的方法得到该光源下 vig 对应的 LSC 参数。

(c) 对 LSC 参数进行 damp，用前一帧的 LSC 参数和当前帧的 LSC 参数进行加权和确定

最终的 LSC 参数，权值为 AwblIRDampCoef（计算白平衡的校正参数也用到这个参数）。

3.2.2 LSC 参数说明

LSC 模块全称：Lens Shading Correction。

LSC 参数包含的 cell 个数 = 分辨 x 光源个数 x 补偿程度个数。

RK IQ 一般调试 2 个分辨率，4 种光源，100%和 70%的两种补偿程度，那么就有 $2 \times 4 \times 2 = 16$ 个 cell 要填写。

默认 xml 文件中，我们设定是 7 种光源，2 个分辨率，2 种补偿程度的 28 个 cell。

可根据调试的光源将相应 lsc 补偿参数填写入相应位置即可。

新增分辨率，光源或者补偿程度，请自行修改增加相应的 cell 信息。

Name: widthxheight_光源名称_补偿程度。

Resolution: widthxheight。

Illumination: 光源名称。

LSC_sectors: LSC 分为 16×16 的分区，产生 17×17 的网格点。

LSC_No: 实际未用到参数，请保持原值。

LSC_Xo: LSC 17×17 网格点 width 方向进行双线性插值系数预先计算所需参数。

LSC_Yo: LSC 17×17 网格点 height 方向进行双线性插值系数预先计算所需参数。

LSC_SECT_SIZE_X: width/2 分成 8 段值，因为对称关系取一半，8 段加起来必须等于 width/2。

LSC_SECT_SIZE_Y: height/2 分成 8 段值，因为对称关系取一半，8 段加起来必须等于 height/2。

Vignetting: 补偿程度。

LSC_SAMPLES_red、LSC_SAMPLES_greenR、LSC_SAMPLES_greenB、LSC_SAMPLES_blue:

分别代表 r, gr, gb, b 四通道 lsc 补偿系数，矩阵为 17×17 。

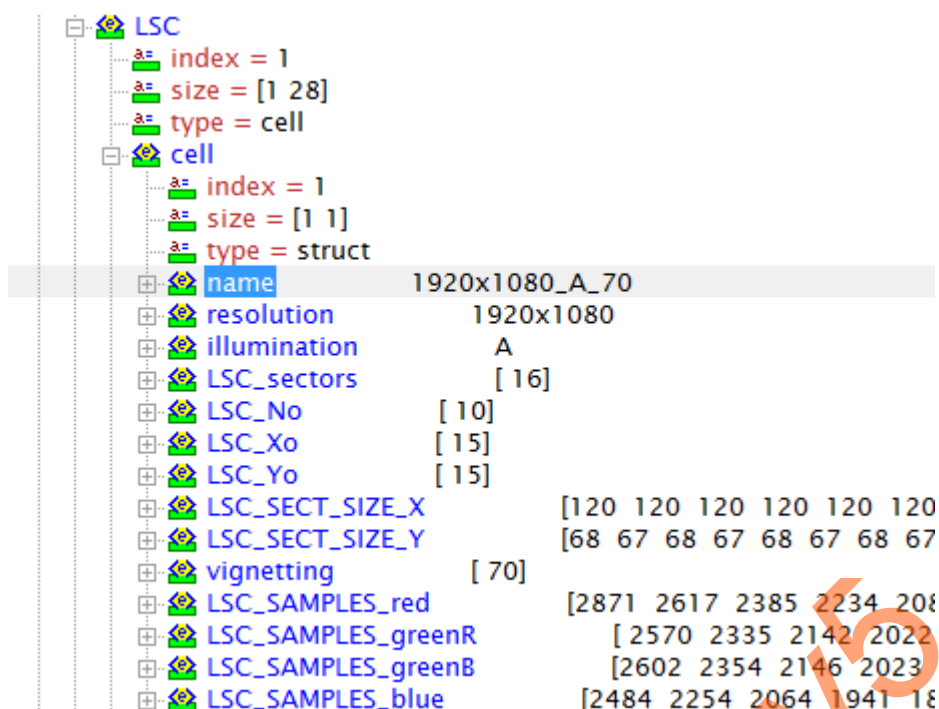
备注：

ISP10 和 ISP11:

补偿系数为 12bit，2bit 整数，10bit 小数，取值范围为 4095-1024，最大补偿倍数为 4 倍。

ISP12:

补偿系数为 13bit，3bit 整数，10bit 小数，取值范围为 8191-1024，最大补偿倍数为 8 倍。



3.2.3 CC 参数说明

CC 参数总个数= 光源个数 X 饱和度个数

Rk IQ 调试一般选择 4 个光源（A、CWF、TL84、D65），两种饱和度（100%， 74%）。

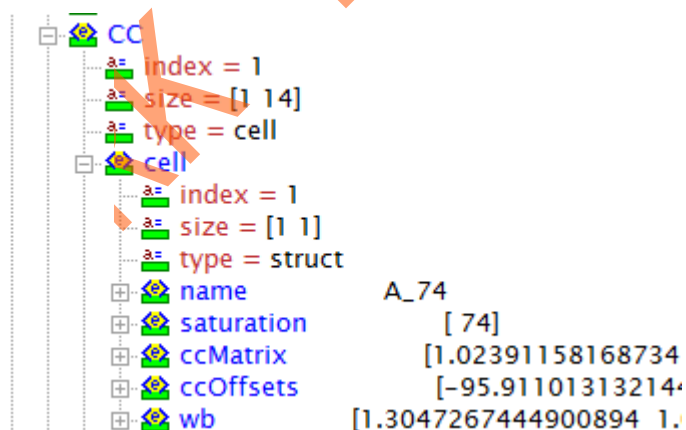
Name: 命名规则，光源名称_饱和度数值。

Saturation: 饱和度数值。

ccMatrix: 色彩校正矩阵，tuning ccm 过程会产生此参数。

ccOffsets: 色彩校正矩阵 offset，tuning ccm 过程可选择是否需要 offset。

Wb: 白平衡校正参数，tuning ccm 过程会产生此参数。

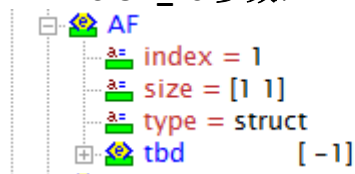


3.2.4 AF 参数说明

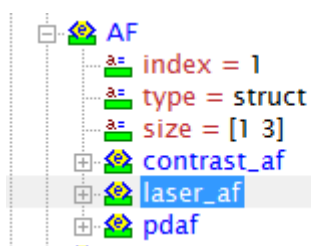
目前 AF 参数有两个版本，版本适配情况如下：

AWB 版本	软件适配情况
AF VERSION_10	CameraHal00
AF VERSION_11	camera_engine_cifisp camera_engine_rkisp

AF VERSION_10 参数：



AFVERSION_11 参数：



contrast_af: 连续对焦

Enable: 使能位，0 关闭，1 打开。

laser_af: 激光对焦

Enable: 使能位，0 关闭，1 打开。

vcmDot: 马达对焦值。

distanceDot: 与马达对焦值对应的距离。

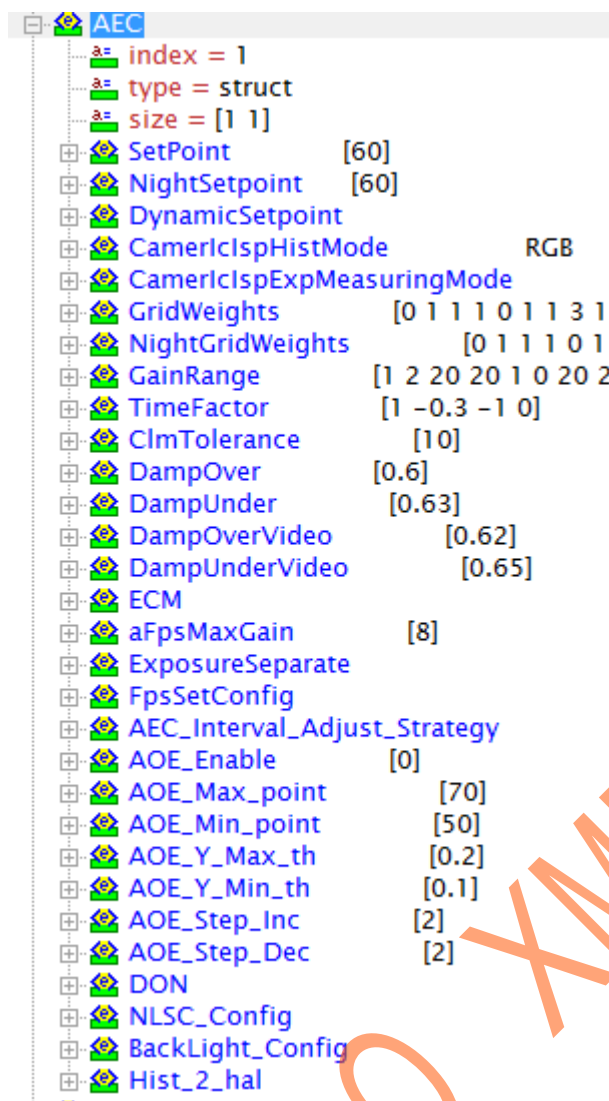
Pdaf: 相位对焦

Enable: 使能位，0 关闭，1 打开。

3.2.5 AEC 参数说明

AEC 模块全称：Auto Exposure Control。

AEC 参数包含了目标值，权重矩阵，gain 转换公式，time 转换公式，aeDamp 系数，曝光分解方式，固定帧率曝光控制，曝光控制策略，夜晚模式判断，支持 normal 和 night 2 种模式参数独立设置。所以此块系数必须填写好，否则对整体曝光影响非常大。



3.2.5.1 AEC 基本参数

AEC 模块中有部分参数以及功能在软件支持上有区别，具体情况如下：

Camera 软件名称	支持参数
Camera_engine_cifisp	All
Camera_engine_rkisp CameraHal00	以下参数不支持： NightSetpoint DynamicSetpoint NightGridWeights FpsSetConfig AEC_Interval_Adjust_Strategy DON BackLight_Config NLSC_Config Hist_2_hal

SetPoint: 曝光目标值。统计是在 demosaic 后 gamma 之前。

NightSetpoint: 夜晚曝光目标值。前面 setpoint 为 normal 模式，NightSetpoint 为 night 模式。

DynamicSetpoint:

```
<DynamicSetpoint index="1" type="cell" size="[1 2]">
  <cell index="1" type="struct" size="[1 1]">
    <name index="1" type="char" size="[1 6]">normal</name>
    <Enable index="1" type="double" size="[1 1]">[1]</Enable>
    <ExposueValue index="1" type="double" size="[1 6]">[0 0.06 0.15 0.5 0.95 1.0]</ExposueValue>
    <DySetpoint index="1" type="double" size="[1 6]">[ 58 58 55 50 35 35]</DySetpoint>
  </cell>
  <cell index="2" type="struct" size="[1 1]">
    <name index="1" type="char" size="[1 6]">night</name>
    <Enable index="1" type="double" size="[1 1]">[1]</Enable>
    <ExposueValue index="1" type="double" size="[1 6]">[0 0.06 0.15 0.5 0.9 1.0]</ExposueValue>
    <DySetpoint index="1" type="double" size="[1 6]">[ 40 40 40 35 35 35]</DySetpoint>
  </cell>
</DynamicSetpoint>
```

动态目标值设置，跟随曝光值变化。

Name: 模式名称，目前只支持 normal 和 night 2 种模式。

Enable: 该模式 dynamic setpoint 功能是否开启，0：关闭，1：开启。

ExposueValue: 最大曝光量的百分比，对应不同曝光量下曝光目标值改变。

DySetpoint: 动态曝光目标值，与上面 ExposueValue 一一对应。

CameraIcspHistMode: histogram 测量模式，模式有五种：R,G,B,RGB,Y。

CameraIcspExpMeasuringMode: ae mean luma 测量模式，模式只有两种：Y 和 RGB。

GridWeights: normal 模式下 ae 分区权重。

NightGridWeights: night 模式下 ae 分区权重。

备注：

ISP10 和 ISP11：ae 分 5x5 区，每个区域权重系数值范围：0-16。

ISP12：aec 分 9x9 区，每个区域权重系数值范围：0-32。

Camera 软件名称	GridWeights 配置方式
Camera_engine_cifisp	根据 ISP10/ISP11 和 ISP12 规则单独配置
CameraHal00	根据 ISP10/ISP11 和 ISP12 规则单独配置
Camera_engine_rkisp	统一采用 ISP12 方式

3.2.5.2 GainRange 、TimeFactor

Gain 和 time 的理论值转换到寄存器值的转换公式。

Camera 软件名称	支持情况
Camera_engine_cifisp	Y
Camera_engine_rkisp	
CameraHal00	N

gain 转换成 reg 值的公式。

转换公式为：set to driver reg = (gain*C1 - C0) / M0 + 0.5

```
<GainRange index="1" type="double" size="[4 7]">
  1.0 2.0 128.0 0.0 1.0 128 255
  2.0 4.0 64.0 -248.0 1.0 376 504
  4.0 8.0 32.0 -756.0 1.0 884 1012
  8.0 16.0 16.0 -1784 1.0 1912 2040]
</GainRange>
```

xml 参数对应:

第一列: gain 区间起始值,

第二列 gain 区间结束值,

第三列: C1,

第四列: C0,

第五列: M0,

第六列: gain 起始值对应 reg,

第七列: gain 结束值对应 reg。

TimeFactor: time 值转化成驱动 reg 值公式。

转换公式: $\text{reg} = C0 \cdot \text{VTS} + C1 + C2 \cdot (\text{time} \cdot \text{pclk} / \text{HTS} + C3)$

```
<TimeFactor index="1" type="double" size="[1 4]">
  [0.0 0.0 1.0 0.5]
</TimeFactor>
```

xml 对应参数:

第一个: C0 ,

第二个: C1,

第三个 C2,

第四个 C3

根据公式, 反过来计算曝光时间为:

$\text{Time} = ((\text{reg} - C0 \cdot \text{vts} - C1) / C2 - C3) \cdot \text{hts} / \text{pclk}$

3.2.5.3 ClmTolerance 、Damp

ClmTolerance: 目标亮度值容忍度。

实际值曝光区间为 $[\text{setpoint} - \text{setpoint} \times \text{ClmTolerance}, \text{setpoint} + \text{setpoint} \times \text{ClmTolerance}]$ 。

Damp 系数: 用来调整 ae 调节速度, 当前曝光值与上一帧曝光值加权平均。

最终曝光值 = 当前曝光值 \times DampCoef + 上一帧曝光值 $\times (1 - \text{DampCoef})$

为保证 aec 调节速度和平滑过度, damp 系数一般取值区间为 $[0.5, 0.8]$ 。

DampOver: 当亮度值高于 setpoint, aec Damp 系数。

DampUnder: 当亮度值低于 setpoint, aec Damp 系数。

DampOverVideo: video 模式下, 当亮度值高于 setpoint, aec Damp 系数。

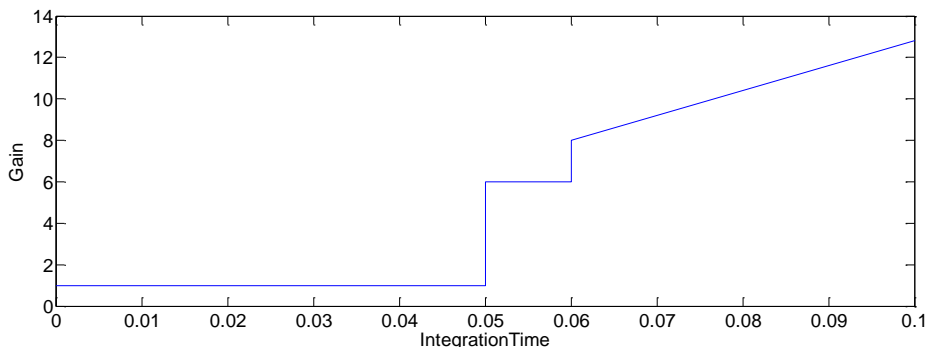
DampUnderVideo: video 模式下, 当亮度值低于 setpoint, aec Damp 系数。

ECM: 只需要修改相应分辨率即可, 其他参数均未被使用。

aFpsMaxGain: 目前此参数也未被使用。

3.2.5.4 ExposureSeparate

如下图，定义 6 个点曝光值分解区间进行分解。



如图：

每个区间固定 time 不变 gain 变化，或者固定 gain 不变 time 变化。

最后一段区间由最后两个点定义变化，最后一个点为最大曝光时间和曝光 gain 值。

TimeDot: 如上图六个点对应 time 值，time 单位为 s

GainDot: 如上图六个点对应 gain 值。

1: HAL00 代码曝光分解参数如下

```
<EcmDotEnable index="1" type="double" size="[1 1]">
[0]
</EcmDotEnable>
<EcmTimeDot index="1" type="double" size="[1 6]">
[0 0.03 0.03 0.03 0.03 0.03]
</EcmTimeDot>
<EcmGainDot index="1" type="double" size="[1 6]">
[1 1 5 5 8 16]
</EcmGainDot>
```

Hal00 代码有曝光自动分解公式在 aec 代码里。

后面再 IQ 中增加了曝光分解设置，通过 EcmDotEnable 使能。

建议一个分辨率可以使能曝光分解。如果 sensor 有多个分辨率，不建议使用此功能。

2: Rkisp 和 cifisp 代码曝光分解参数如下：

Rkisp 和 cifisp 参数支持 2 个模式设置曝光分解设置，

仅 camera_engine_cifisp 上可使能两种模式，rkisp 芯片默认使用 normal 模式。

```
<ExposureSeparate index="1" type="cell" size="[1 2]">
<cell index="1" type="struct" size="[1 1]">
<name index="1" type="char" size="[1 6]">normal</name>
<TimeDot index="1" type="double" size="[1 6]">[0 0.03 0.03 0.04 0.04 0.04]</TimeDot>
<GainDot index="1" type="double" size="[1 6]">[1 1 5 5 8 16]</GainDot>
</cell>
<cell index="2" type="struct" size="[1 1]">
<name index="1" type="char" size="[1 5]">night</name>
<TimeDot index="1" type="double" size="[1 6]">[0 0.03 0.03 0.04 0.04 0.04]</TimeDot>
<GainDot index="1" type="double" size="[1 6]">[1 1 5 5 8 32]</GainDot>
</cell>
</ExposureSeparate>
```

如果客户在更新代码中发现 IQ 中没有此项选项，请将旧 xml 中 timedot 和 gainDot 复制到两种模式下即可。

如果客户希望使用两种模式采用不同曝光分解测量，请根据 IQ xml 参数设置说明书重新设

置 night 模式下的参数,并将旧 IQ xml 中的 TimeDot 和 GainDot 参数复制到新 IQ 中的 normal 模式下。

FpsSetConfig: 固定帧率曝光时间分解控制,

```
<FpsSetConfig index="1" type="struct" size="[1 1]">
  <FpsSet_enable index="1" type="double" size="[1 1]">
    [1]
  </FpsSet_enable>
  <isFpsFix index="1" type="double" size="[1 1]">
    [1]
  </isFpsFix>
  <FpsFix_TimeDot index="1" type="double" size="[1 6]">
    [0 1.0 1.0 1.0 1.0 1.0]
  </FpsFix_TimeDot>
</FpsSetConfig>
```

FpsSet_enable: 是否打开此功能, 0: 关闭, 1: 打开。

isFpsFix: 是否是固定帧率, 0: 不是固定帧率, 1: 固定帧率。

FpsFix_TimeDot: 固定帧率采用曝光时间分解点值,按照固定帧率最大曝光时间百分比来填。替换上面 ExposureSeparate 中 TimeDot。

Aec Antiflicker 的方式:

曝光分解过程中, 对 flicker 进行调节方式是:

凡是曝光时间大于 flicker 时间的, 必须为 flicker 整数倍。

凡是曝光时间小于 flicker 时间的, 不为 flicker 整数倍。

另外此曝光分解策略同时定义了最大曝光时间和最大曝光 gain。

想改变帧率的, 请在这里调节最大曝光时间。

想改变最大 gain 的, 请在这里调节最大 gain 值, 但不能超过驱动设置最大值。

驱动最大值由 GainRange 里决定。

3.2.5.5 AEC_Interval_Adjust_Strategy

曝光控制调节策略

```
<AEC_Interval_Adjust_Strategy index="1" type="struct" size="[1 1]">
  <enable index="1" type="double" size="[1 1]">[1]</enable>
  <dLuma_high_th index="1" type="double" size="[1 1]">[0.5]</dLuma_high_th>
  <dLuma_low_th index="1" type="double" size="[1 1]">[0.03]</dLuma_low_th>
  <adjust_trigger_frame index="1" type="double" size="[1 1]">[100]</adjust_trigger_frame>
</AEC_Interval_Adjust_Strategy>
```

Enable: 是否使用曝光控制策略, 0: 关闭, 1: 开启。

dLuma_high_th: 当两帧图像亮度差大于等于此值时, 立即进行曝光调整。

dLuma_low_th: 当两帧图像亮度差大于等于此值时, 则认为场景曝光不稳定, 等待曝光稳定再进行调节。

adjust_trigger_frame: 当曝光不稳定时候, 统计帧数, 如果其中有相邻两帧曝光差有大于 dLuma_low_th 时候, 则认为曝光不稳定。

3.2.5.6 AOE

AOE: 防过曝调节策略, 功能还未完成, 请保持 AOE_Enable 为 0。

```
<AOE_Enable index="1" type="double" size="[1 1]">
  [0]
</AOE_Enable>
<AOE_Max_point index="1" type="double" size="[1 1]">
  [70]
</AOE_Max_point>
<AOE_Min_point index="1" type="double" size="[1 1]">
  [50]
</AOE_Min_point>
<AOE_Y_Max_th index="1" type="double" size="[1 1]">
  [0.2]
</AOE_Y_Max_th>
<AOE_Y_Min_th index="1" type="double" size="[1 1]">
  [0.1]
</AOE_Y_Min_th>
<AOE_Step_Inc index="1" type="double" size="[1 1]">
  [2]
</AOE_Step_Inc>
<AOE_Step_Dec index="1" type="double" size="[1 1]">
  [2]
</AOE_Step_Dec>
```

3.2.5.6 DON

DON: 夜晚模式判断

```
<DON index="1" type="struct" size="[1 1]">
  <NightTrigger index="1" type="double" size="[1 1]">
    [0]
  </NightTrigger>
  <NightMode index="1" type="double" size="[1 1]">
    [1]
  </NightMode>
  <DON_Day2Night_Fac_th index="1" type="double" size="[1 1]">
    [7]
  </DON_Day2Night_Fac_th>
  <DON_Night2Day_Fac_th index="1" type="double" size="[1 1]">
    [15]
  </DON_Night2Day_Fac_th>
  <DON_Bouncing_th index="1" type="double" size="[1 1]">
    [100]
  </DON_Bouncing_th>
</DON>
```

此模块是软件上判断白天和夜晚的一些参数。

以前软件判断比较简单, 后面有进行升级。一些参数含义重新定义或者删除。

Night_Trigger: 用于配置模式切换所依赖的触发条件。

- 0: 不启用 Night 模式(TRIGGER_OFF)
- 1: 依赖感光元件(LIGHT_SENS)
- 2: 依赖曝光参数(NO_LIGHT_SENS)

Night_Mode: 用于配置进入夜晚模式后是否开启黑白模式。

- 1: 正常模式(NORMAL)
- 2: 黑白模式(WHITE_BLACK)

当 **Night_Trigger=2** 时, AEC 模块将依赖以下公式与参数决定使用白天或黑夜模式

公式: $\text{Factor} = (\text{MeanLuma} / (\text{Gain} \times \text{Time} \times 10))$

其中 MeanLuma 为亮度加权平均值, Time 的单位为 s。

DON_Bouncing_th: 稳定帧数

DON_Day2Night_Fac_th: 处于白天模式时, Factor 小于该值且不间断持续大于稳定帧数时切至黑夜模式

DON_Night2Day_Fac_th: 处于黑夜模式时, Factor 大于该值且不间断持续大于稳定帧数时切至白天模式。

3.2.5.7 NLSC_Config(无光敏电阻红外与可见光切换方案)

NLSC_Config: 无光敏电阻红外与可见光切换方案。

Enable: 使能位。 0: 不使能 1: 使能

IR_rg: 环境光仅有红外光下, 白平衡统计对应 r 值。

IR_bg: 环境光仅有红外光下, 白平衡统计对应 b 值。

Max_Dis: 可见光的几种标准光源, 与仅有红外光白平衡点最大距离。

Color2Black_count: 彩色切换黑白, 连续统计低于彩色切换黑白阈值帧数后, 才从彩色模式切换到黑白模式。此值就是连续多少帧设定。

Color2Black_thresh: 当可见光环境亮度较低时, 会打开红外灯, 撤掉 IRcut 滤光片, 捕捉红外图像, 并将显示从彩色模式切换黑白模式。此值就是彩色模式切换到黑白模式阈值, 与图像亮度和曝光相关, $\text{Color2Black_thresh} = \text{MeanLuma} / (\text{Gain} \times \text{Time})$ 。

Black2Color_count: 在红外模式下, 当可见光亮度提高大于阈值, 并且连续帧率超过 Black2Color_count 值, 则将红外灯关闭, IRcut 滤光片覆盖在 camera 上, 只采集可见光图像。并将显示从黑白模式切换到彩色模式。

Black2Color_thresh: $\text{Black2Color_thresh} = \text{MeanLuma} / (\text{Gain} \times \text{Time})$ 。当图像亮度和曝光达到一定程度后, 会去判断是否要将 IRcut 滤光片重新覆盖上, 只采集可见光图像。

Black2Color_vb_percent: 当图像白平衡判断到可见光占全部光源一定比例后, 会去判断是否要将 IRcut 滤光片重新覆盖上, 只采集可见光图像。

Color2Black_stable_fps: 当环境亮度较低, 打开红外灯后, 需要稳定一段时间, 等待曝光稳定。此值为等待曝光稳定帧数。

还需配合 DON 夜晚模式切换设定, 如下:

Night_Trigger: 必须设置为 2, 依赖曝光参数(NO_LIGHT_SENS)

Night_Mode: 必须设置为 1, 黑白模式(WHITE_BLACK)

3.2.5.8 BackLight_Config

BackLight_Config: 背光参数调节。

```
<BackLight_Config>
  <Enable index="1" type="double" size="[1 1]">
    [0]
  </Enable>
  <LumaLowTh index="1" type="double" size="[1 1]">
    [100]
  </LumaLowTh>
  <LumaHighTh index="1" type="double" size="[1 1]">
    [180]
  </LumaHighTh>
  <WeightMinTh index="1" type="double" size="[1 1]">
    [0.1]
  </WeightMinTh>
  <WeightMaxTh index="1" type="double" size="[1 1]">
    [1.0]
  </WeightMaxTh>
</BackLight_Config>
```

Aec 分区的亮度，当分区的亮度较高时，降低高亮度分区的权重，来提高 aec。

通过亮度区间线性插值来逐渐降低高

亮度的权重。越亮，权重越低。

Enable: 模块使能位，1: 使能， 0: 关闭

LumaLowTh: 亮区权重插值区间最低亮度值

LumaHighTh: 亮区权重插值区间最高亮度值

WeightMinTh: 亮区权重最低值

WeightMaxTh: 亮区权重最高值

3.2.5.9 Hist_2_hal

```
<Hist_2_hal>
  <Enable index="1" type="double" size="[1 1]">
    [0]
  </Enable>
  <LowHistBinTh index="1" type="double" size="[1 1]">
    [13]
  </LowHistBinTh>
</Hist_2_hal>
```

此参数主要是 RV1108 DSP 人脸识别算法需要用的参数。

主要作用是将过曝的几个 hist 的百分比传给应用层。

Enable: 模块使能位，1: 使能， 0: 关闭

LowHistBinTh: 从第几个 bin 开始为过曝 hist，累加计算到最大 bin 的过曝百分比传给上层应用。

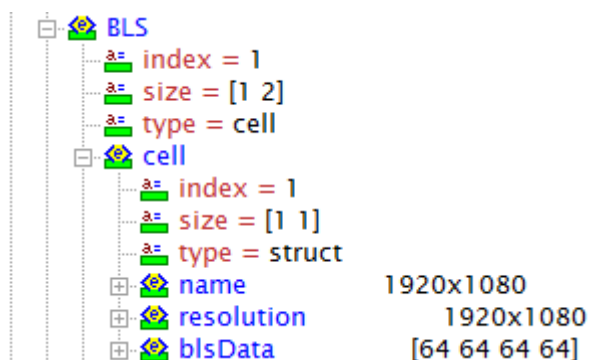
3.2.6 BLS 参数说明

BLS: isp 中固定减模式，此参数为 **12bit**。不同分辨率需对应不同 cell。

Name: width x height

Resolution : width x height

blsData: 12bit, 参数依次为 r、gr、gb、b 通道相应的黑电平值。



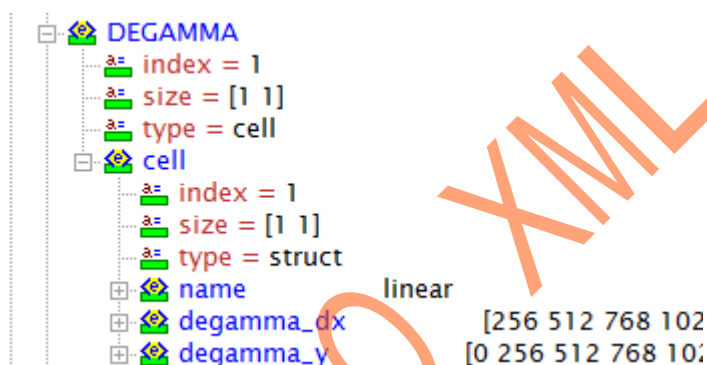
3.2.7 DEGAMMA 参数说明

该模式尚未支持，不建议使能；

Name: 模式选择

degamma_dx: degamma 曲线 x 轴点间距。

degamma_y: degamma 曲线 y 轴点



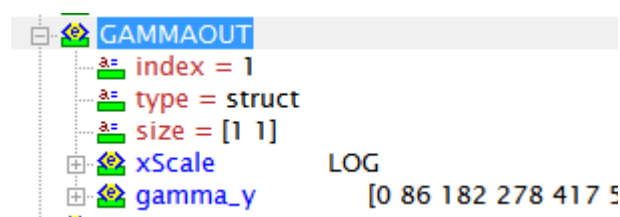
3.2.8 GAMMAOUT/GOC 参数说明

Android HAL00 代码此模块对应名称为 GAMMAOUT，

除 HAL00 外，此模块对应名称为 GOC。

两者是相同的模块功能，只是代码差异导致名称不一样而已。

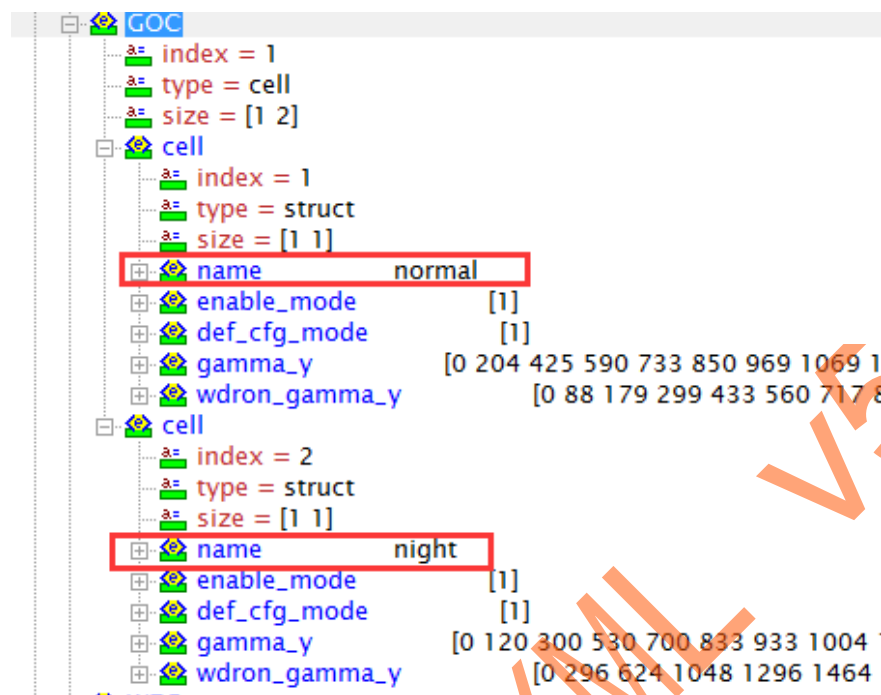
3.2.8.1 GAMMAOUT



xScale: 选择 gamma 曲线 x 轴对应的模式, LOG: 非均匀等分, EQU: 均匀等分。

gamma_y: gammaout 曲线 y 轴对应点坐标。

3.2.8.2 GOC



GOC 模块全称: Gamma Out Curve, Gamma out 曲线配置。

目前此模块支持 **normal** 和 **night** 两种模式独立设置, 仅 **camera_engine_cifisp** 上可启用。

其他版本 ISP 代码默认使用 **normal** 模式的 gamma 相关参数。

每个模式下设置为:

enable_mode: gamma out 是否打开, 0 关闭, 1 打开。

def_cfg_mode: gamma out 模式设置, 0 关闭, 1 为 setting, 2 为 default 即 gamma2.2

gamma_y: wdr 关闭时, 所使用 gamma out 曲线输出对应坐标。

wdron_gamma_y: wdr 打开时, 所使用 gamma out 曲线输出对应坐标。

Gamma_Y 总共 34 个点。第 34 个点必须为 0。

内部 gamma_x 坐标为 LOG 模式, 非均匀等分。

RK3288 等平台 isp10 取值范围为 0-1023, 实际只用到前 17 个点, 其余补零。

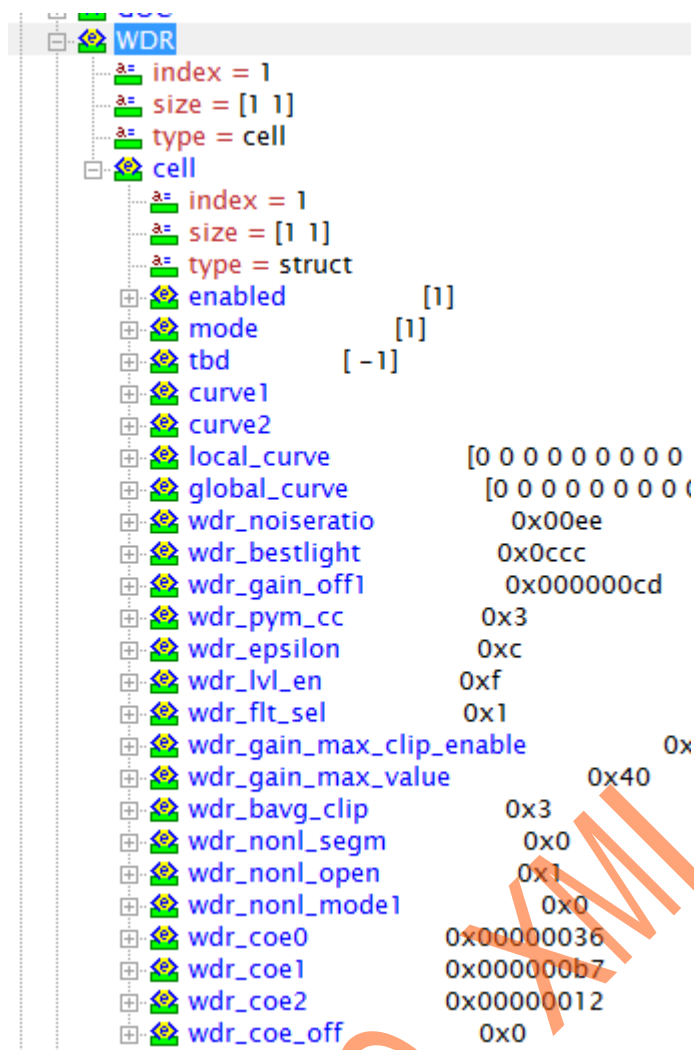
RK1108 平台 isp11 取值范围为 0-4095, 实际只用到前 33 个点, 最后一个补零。

RK3326 平台 isp12 取值范围为 0-4095, 实际只用到前 33 个点, 最后一个补零。

3.2.9 WDR 参数说明

WDR 模块全称: Wide Dynamic Range

WDR 模块只有在 **rk1108** 上的 **isp11**, 以及 **rk3326** **isp12** 中有相应硬件支持, **ISP10** 平台暂时未有此功能, 请注意关闭模块功能。



Enable: 模块打开关闭控制，1 为打开， 0 为关闭。

Mode: 模式切换开关，wdr 目前有两种方式，

0 为：global 模式，根据直方图统计，改变全局 gamma 曲线形状。

1 为：block 模式，将图像分区，每个区域有自己相应 gamma 曲线调节动态范围。

curve1: 目前无实际意义保留该项目

Curve2: 目前无实际意义保留该项目

local_curve: block 模式下，非线性转化曲线。全为 0 默认为使用代码中默认值，如果知道实际意义，可从这设置自己相应的曲线。实际只使用前 16 段，剩余补零。

global_curve: global 模式下，wdr 全局算法映射曲线设置，全为 0 默认为使用代码中默认值，如果知道实际意义，可从这设置自己相应的曲线。使用 33 段。

剩余 wdr_noiseratio、wdr_bestlight、wdr_gain_off1、wdr_pym_cc、wdr_epsilon、wdr_lvl_en、wdrflt_sel、wdr_gain_max_clip_enable、wdr_gain_max_value、wdr_bavg_clip、wdr_nonl_segm、wdr_nonl_open、wdr_nonl_mode1、wdr_coe0、wdr_coe1、wdr_coe2、wdr_coe_off:

均为 wdr 相关寄存器设置，请保持目前使用的 xml 中的默认值。

如需修改请参考 wdr 的相关文档进行设定。

这些参数设置均为 16 进制寄存器值，使用 0x 开头。

```
<wdr_maxgain_filter index="1" size="[1 2]" type="struct">
  <wdr_maxgain_filter_enable index="1" size="[1 1]" type="double">
    [1]
  </wdr_maxgain_filter_enable>
  <wdr_sensor_gain_level index="1" size="[1 5]" type="double">
    [1 2 4 8 16]
  </wdr_sensor_gain_level>
  <wdr_maxgain_level index="1" size="[1 5]" type="double">
    [4 3 2 2 2]
  </wdr_maxgain_level>
</wdr_maxgain_filter>
```

新增 wdr MaxGain 控制，此控制跟随曝光的 gain 值进行变化。

这里参数皆为 double 类型，已不是 16 进制寄存器值，无需 0x 开头。

wdr_maxgain_filter_enable: MaxGain 控制是否打开。0 关闭，1 打开。

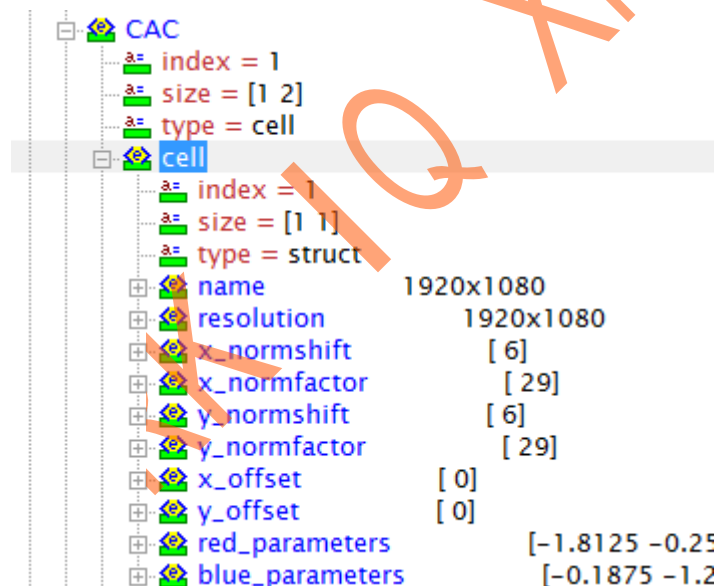
wdr_sensor_gain_level: 曝光 gain 档位。目前设定为 5 档，可进行增加或减少，增加或者减少时 size 要跟随进行修改。

wdr_maxgain_level: Maxgain 跟随上面 gain 值进行改变，档位数必须和上面的 wdr_sensor_gain_level 档位数相同，size 相同。Maxgain 取值范围为[0, 15]

3.2.10 CAC 参数说明

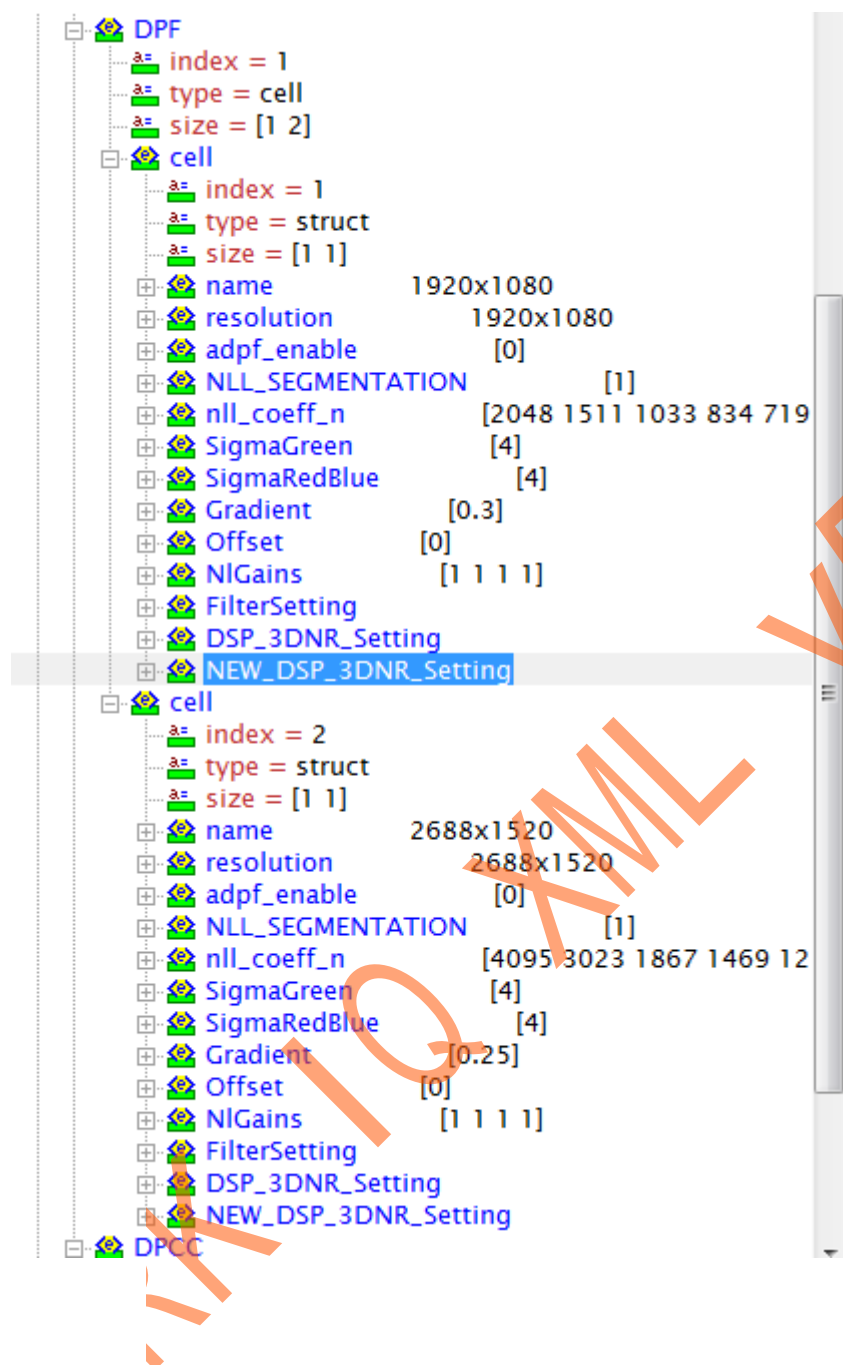
CAC 模块全称：Chromatic Aberration Correction。

目前建议关闭 CAC 校正功能，参数保留原来值。



index	1
size	[1 2]
type	cell
cell	
index	1
size	[1 1]
type	struct
name	1920x1080
resolution	1920x1080
x_normshift	[6]
x_normfactor	[29]
y_normshift	[6]
y_normfactor	[29]
x_offset	[0]
y_offset	[0]
red_parameters	[-1.8125 -0.25]
blue_parameters	[-0.1875 -1.2]

3.2.11 DPF 参数说明



3.2.11.1 DPF

模块全称: denoise pre-filter, 作用是在 demosaic 前对 raw 进行一次去噪。

Name: 分辨率名称 width x height。

resolution: 分辨率名称 width x height。

adpf_enable: 是否打开 adpf 模块, 0 为关闭, 1 为打开。

NLL_SEGMENTATION: adpf 噪声方差曲线 x 间距模式, 0 为等间距, 1 为非均匀间距。

nll_coeff_n: adpf 噪声方差曲线 x 轴点对应 y 轴值。

SigmaGreen: Gr 和 Gb 通道去噪半径, 代码已固定为 4, 无需改变此参数。

SigmaRedBlue: r 和 b 通道去噪半径，可选两种，1 种为 4，1 种为 3。

Gradient 、Offset: 与曝光 gain 值相关的计算去噪力度的参数。

公式: $fStrength = \sqrt{fGradient * fSensorGain} + fOffset$;

根据算法部门的研究，Gradient 可根据 IQ tuning Dpf 模块的图进行求取。

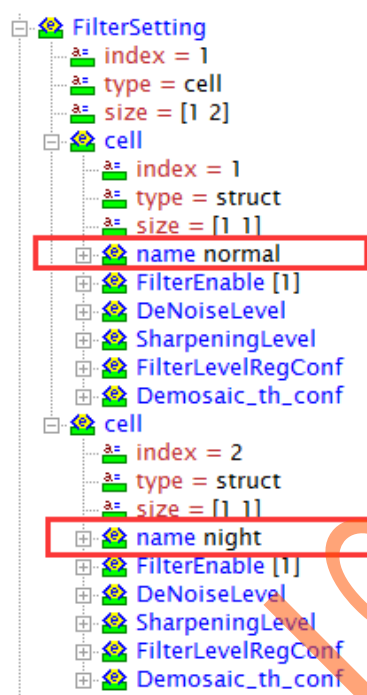
假设最大方差为 3，对应 nll_coeff_nl 参数最小值 355。

$Gradient = (355 * 3.9 * 3 / 255 / 32)^2 = 0.25$ ，式中除 3 和 355 为变量外，其余为固定值。

NlGains: 代表顺序为 R、GR、GB、B 通道其他模块 gain 值影响。一般保持 1 不变。

3.2.11.2 FilterSetting

去马赛克 denoise 和 sharp filter 的控制参数也定义在 adpf 模块中



每个分辨率下 FilterSetting 支持 2 种模式分别配置，normal 和 night 模式。

仅 camera_engine_cifisp 上支持这 2 种模式，其他平台默认使用 normal 模式参数。

每个模式下设置为：

FilterEnable: sharp 和 denoise 的控制是否打开。 0: 关闭，1: 打开。

DeNoiseLevel:

```
<DeNoiseLevel index="1" type="struct" size="[1 2]">
  <gains index="1" type="double" size="[1 6]">[1 1.1 6 8 10 12]</gains>
  <dlevel index="1" type="double" size="[1 6]">[0 1 1 1 1 1]</dlevel>
</DeNoiseLevel>
```

Gains: 曝光 gain 等级，去噪跟随曝光 gain 值进行变化。

Dlevel: 去噪力度等级设置。取值范围[0, 10]。

不同 gain 之间， $g < (g1 + g2) / 2$ ，采用 $g1$ 对应的去噪等级。

$g > (g_1 + g_2) / 2$, 采用 g_2 对应的去噪等级。

SharpeningLevel:

```
<SharpeningLevel index="1" type="struct" size="[1 2]">
  <gains index="1" type="double" size="[1 6]">[1 1.1 6 8 10 12]</gains>
  <slevel index="1" type="double" size="[1 6]">[3 2 2 2 2 2]</slevel>
</SharpeningLevel>
```

Gains: 曝光 gain 等级, 锐化跟随曝光 gain 值进行变化。

Slevel: 锐化力度等级设置。取值范围[0, 10]。

不同 gain 之间, $g < (g_1 + g_2) / 2$, 采用 g_1 对应的锐化等级。

$g > (g_1 + g_2) / 2$, 采用 g_2 对应的锐化等级。

3.2.11.3 FilterLevelRegConf

配置 filter 不同等级对应寄存器设置。

原本 filter 不同等级对应寄存器设置是内部代码设置固定的。

此参数就是让我们可以设置不同等级对应的寄存器设置。

```
<FilterLevelRegConf index="1" type="struct" size="[1 14]">
  <FilterLevelRegConfEnable index="1" type="double" size="[1 1]">[1]</FilterLevelRegConfEnable>
  <FilterLevel index="1" type="double" size="[1 5]">[0 1 2 3 4]</FilterLevel>
  <flt_chr_h_mode index="1" type="double" size="[1 5]">[0 3 3 3 3]</flt_chr_h_mode>
  <flt_chr_v_mode index="1" type="double" size="[1 5]">[1 3 3 3 3]</flt_chr_v_mode>
  <flt_grn_stage1 index="1" type="double" size="[1 5]">[7 6 6 5 5]</flt_grn_stage1>
  <flt_thresh_bl0 index="1" type="double" size="[1 5]">[0 0 8 13 23]</flt_thresh_bl0>
  <flt_thresh_bl1 index="1" type="double" size="[1 5]">[0 0 2 5 10]</flt_thresh_bl1>
  <flt_fac_bl0 index="1" type="double" size="[1 5]">[0 0 2 4 6]</flt_fac_bl0>
  <flt_fac_bl1 index="1" type="double" size="[1 5]">[0 0 0 0 2]</flt_fac_bl1>
  <flt_thresh_sh0 index="1" type="double" size="[1 5]">[0 20 20 20 20]</flt_thresh_sh0>
  <flt_thresh_sh1 index="1" type="double" size="[1 5]">[0 80 80 80 80]</flt_thresh_sh1>
  <flt_fac_sh0 index="1" type="double" size="[1 5]">[8 8 10 16 15]</flt_fac_sh0>
  <flt_fac_sh1 index="1" type="double" size="[1 5]">[10 10 13 32 15]</flt_fac_sh1>
  <flt_fac_mid index="1" type="double" size="[1 5]">[4 4 4 15 8]</flt_fac_mid>
</FilterLevelRegConf>
```

FilterLevelRegConfEnable:

是否改变原有等级内部寄存器设置, 0: 不改变, 1: 改变并采用下面设置。

FilterLevel:

需要改变的默认等级配置。如上图, 会改变 0 1 2 3 4 对应等级的内部原本设置的寄存器值。等级取值范围[0 10]。

flt_chr_h_mode: RB 通道水平方向去噪设置, 取值范围[0 3]。

flt_chr_v_mode: RB 通道垂直方向去噪设置, 取值范围[0 3]。

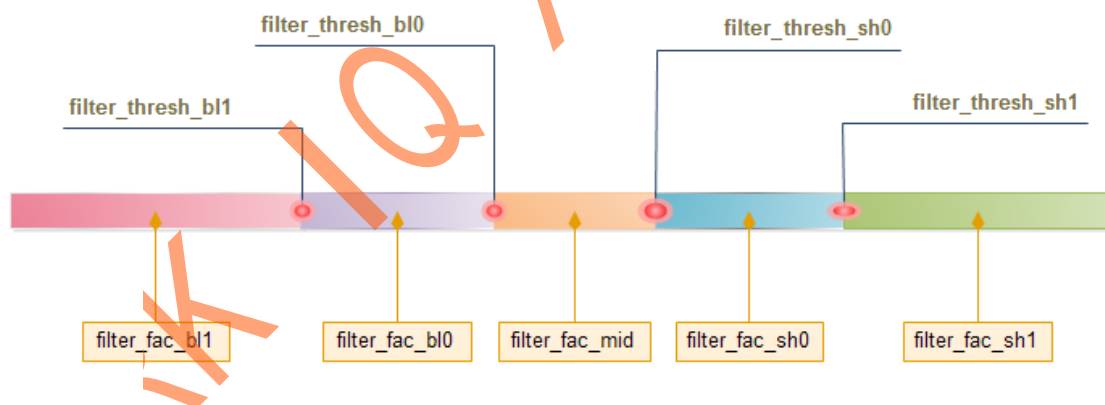
flt_grn_stage1: G 通道低通滤波去噪模板等级, 取值范围[0 8]。

Table 4: First filter stage selection

Filter 1 select	Filter matrix	Comment / use case
0	$\begin{bmatrix} 6 & 8 & 6 \\ 8 & 8 & 8 \\ 6 & 8 & 6 \end{bmatrix}$	Max blur filter for very noisy input data or pre filter for down scaling
1	$\begin{bmatrix} 5 & 8 & 5 \\ 8 & 12 & 8 \\ 5 & 8 & 5 \end{bmatrix}$	blur filter for noisy input data
2	$\begin{bmatrix} 4 & 8 & 4 \\ 8 & 16 & 8 \\ 4 & 8 & 4 \end{bmatrix}$	Maximum line noise rejection (like separate vertical + horizontal filter with $\begin{bmatrix} 1 & 2 & 1 \end{bmatrix}/4$ mask)
3	$\begin{bmatrix} 5 & 6 & 5 \\ 6 & 20 & 6 \\ 5 & 6 & 5 \end{bmatrix}$	Maximum line noise rejection with balanced symmetry
4	$\begin{bmatrix} 4 & 6 & 4 \\ 6 & 24 & 6 \\ 4 & 6 & 4 \end{bmatrix}$	Optimum compromise between noise rejection and sharpening
5	$\begin{bmatrix} 3 & 5 & 3 \\ 5 & 32 & 5 \\ 3 & 5 & 3 \end{bmatrix}$	Weak low pass for sharpening
6	$\begin{bmatrix} 3 & 4 & 3 \\ 4 & 36 & 4 \\ 3 & 4 & 3 \end{bmatrix}$	Weak low pass for sharpening
7	$\begin{bmatrix} 2 & 3 & 2 \\ 3 & 44 & 3 \\ 2 & 3 & 2 \end{bmatrix}$	Weakest low pass filter mask for sharpening
8	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 64 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	bypass mask enables 3x3 over all filter modes

Filter G 通道的高通锐化原理

锐化是以梯度和为衡量，四个阈值划分出 5 个区域，分别对每个区域的锐化强度进行配置。下图为梯度和对应阈值和区间锐化强度参数：



flt_thresh_bl0, flt_thresh_bl1, flt_thresh_sh0, flt_thresh_sh1:

梯度和分区间的阈值，这四个阈值，分割出 5 个区间，取值范围[0 1023]。

flt_fac_bl0, flt_fac_bl1, flt_fac_sh0, flt_fac_sh1, flt_fac_mid:

上面 5 个区间对应的锐化强度值，取值范围[0 63]。

3.2.11.4 Demosaic_th_conf

去马赛克阈值设置


```
<Demosaic_th_conf index="1" type="struct" size="[1 1]">
  <gains index="1" type="double" size="[1 6]">[1 1.1 6 8 10 12]</gains>
  <demosaic_th_level index="1" type="double" size="[1 6]">[0 0 0 0 0 0]</demosaic_th_level>
</Demosaic_th_conf>
```

Gains: 随着 gain 变化，可对 demosaic 阈值进行调整。

demosaic_th_level: 阈值调整参数，与上面 gain 一一对应。取值范围：[0 255]。

3.2.11.5 Demosaic_lp_conf

GIC 模块。

The screenshot shows the configuration tree for the Demosaic_lp_conf module. The tree structure is as follows:

- Demosaic_lp_conf
 - index = 1
 - type = struct
 - size = [1 1]
 - lp_en [0]
 - rb_filter_en [1]
 - hp_filter_en [1]
 - use_old_lp [0]
 - lu_divided [240 200 140 80]
 - gainsArray [1 2 4 6 8 16]
 - thH_divided0 [20 20 28 28 39 45]
 - thH_divided1 [16 16 22 22 31 31]
 - thH_divided2 [10 10 14 14 20 25]
 - thH_divided3 [9 9 11 11 15 22]
 - thH_divided4 [6 6 7 7 10 15]
 - thCSC_divided0 [25 25 45 45 48]
 - thCSC_divided1 [20 20 30 30 36]
 - thCSC_divided2 [18 18 20 20 29]
 - thCSC_divided3 [15 15 15 15 14]
 - thCSC_divided4 [8 8 8 8 12 15]
 - diff_divided0 [40 40 56 56 78 78]
 - diff_divided1 [28 28 39 39 55 55]
 - diff_divided2 [18 18 25 25 35 48]
 - diff_divided3 [14 14 20 20 28 43]
 - diff_divided4 [8 8 11 11 15 20]
 - varTh_divided0 [300 300 300 300]
 - varTh_divided1 [100 100 100 100]
 - varTh_divided2 [80 80 80 80 160]
 - varTh_divided3 [50 50 70 70 120]
 - varTh_divided4 [30 30 40 40 70 1]
 - thgrad_r_fct [24]
 - thdiff_r_fct [24]
 - thvar_r_fct [24]

thgrad_b_fct	[24]
thdiff_b_fct	[24]
thvar_b_fct	[24]
similarity_th	[7]
th_var_en	[1]
th_csc_en	[1]
th_diff_en	[1]
th_grad_en	[1]
th_grad	[13]
th_diff	[15]
th_csc	[18]
th_var	[12]
flat_level_sel	[4]
pattern_level_sel	[8]
edge_level_sel	[8]

此模块仅 ISP12 平台有此功能，其他平台未支持，请根据平台注意此模块使能情况。

Demosaic LP 算法说明：

区域划分 lp 算法说明

If (h_v_max_grad < thH & count >= sw_similarity_th & diff_avg < thCSC & var < varTH)

当前点处于平坦区，强滤波；滤波强度可以配置。

else if (h_v_max_grad >= sw_th_grad | count < sw_similarity_th | diff_avg >= sw_th_csc | var > sw_th_var)

当前点处于边缘区，默认不滤波；滤波强度可以配置。

else

当前点处于纹理区，弱滤波；滤波强度可配置。

其中：

h_v_max_grad: 梯度；thH: 平坦区 grad 阈值；sw_th_grad: 边缘区 grad 阈值。

count : 3*3 相似点个数；sw_similarity_th: 平坦区和边缘区相似点个数阈值。

diff_avg: gb-gr 差异；thCSC: 平坦区 diff 阈值；sw_th_csc: 边缘区 diff 阈值。

Var: 方差；varTH: 平坦区 var 阈值；sw_th_var: 边缘区 var 阈值。

Demosaic LP 参数说明：

lp_en: lp 模块开关，0: 关闭，1: 开启。

rb_filter_en: rk demosaic 算法开启，rb 通道 filter 是否开启，0: 关闭，1: 开启。

hp_filter_en: rk 算法锐化是否开启，0: 关闭，1: 开启。

use_old_lp: 是否使用 img 的 lp 算法。0: 关闭，1: 开启。

平坦区阈值条件：

lu_divided: 亮度 0-255 分区。共 4 个点，分成 5 个区间。不同亮度，控制不同阈值。

gainsArray: 不同 gain 下，控制不同的阈值。

thH_divided0, thH_divided1, thH_divided2, thH_divided3, thH_divided4:

5 个亮度区间在不同 gain 下的 thH 阈值控制。

后缀 01234 为对应 5 个亮度区间阈值控制。

thH_dividedx 为每个亮度区间对应上面 gain array 中不同 gain 值时，设置 g 通道对应不同 thH 阈值，取值范围[0-255]。

thCSC_divided0, thCSC_divided1, thCSC_divided2, thCSC_divided3, thCSC_divided4:

5 个亮度区间在不同 gain 下的 thcsc 阈值控制。后缀 01234 为对应 5 个亮度区间阈值控制。
thCSC_dividedx 为每个亮度区间对应上面 gain array 中不同 gain 值时, 设置 g 通道对应不同 thCSC 阈值, 取值范围[0-255]。

diff_divided0, diff_divided1, diff_divided2, diff_divided3, diff_divided4:

5 个亮度区间在不同 gain 下的 diff 阈值控制。

后缀 01234 为对应 5 个亮度区间阈值控制。

diff_dividedx 为每个亮度区间对应上面 gain array 中不同 gain 值时, 设置 g 通道对应不同 diff 阈值, 取值范围[0-255]。

varTh_divided0, varTh_divided1, varTh_divided2, varTh_divided3, varTh_divided4:

5 个亮度区间在不同 gain 下的 var 阈值控制。后缀 01234 为对应 5 个亮度区间阈值控制。

varTh_dividedx 为每个亮度区间对应上面 gain array 中不同 gain 值时, 设置 g 通道对应不同 varTh 阈值, 取值范围[0-65535]。

thgrad_r_fct:

gr 通道基于 g 通道 grad 阈值进行倍数调整, 取值范围[0-127], 默认值 0x18 为 1x 倍数。

thdiff_r_fct:

gr 通道基于 g 通道 diff 阈值进行倍数调整, 取值范围[0-127], 默认值 0x18 为 1x 倍数。

thvar_r_fct:

gr 通道基于 g 通道 var 阈值进行倍数调整, 取值范围[0-127], 默认值 0x18 为 1x 倍数。

thgrad_b_fct:

gb 通道基于 g 通道 grad 阈值进行倍数调整, 取值范围[0-127], 默认值 0x18 为 1x 倍数。

thdiff_b_fct:

gb 通道基于 g 通道 diff 阈值进行倍数调整, 取值范围[0-127], 默认值 0x18 为 1x 倍数。

thdiff_b_fct:

gb 通道基于 g 通道 var 阈值进行倍数调整, 取值范围[0-127], 默认值 0x18 为 1x 倍数。

similarity_th: 算法判断相似点个数阈值, 取值范围[0-8]。

边缘区阈值条件:

th_grad: 边缘区 grad 阈值设定, 取值范围[0-255]。

th_diff: 边缘区 diff 阈值设定, 取值范围[0-255]。

th_csc: 边缘区 csc 阈值设定, 取值范围[0-255]。

th_var: 边缘区 var 阈值设定, 取值范围[0-65535]。

其他寄存器参数说明:

th_var_en: 算法 var 条件是否使能, 0: 不使能, 1: 使能。

th_csc_en: 算法 csc 条件是否使能, 0: 不使能, 1: 使能。

th_diff_en: 算法 diff 条件是否使能, 0: 不使能, 1: 使能。

th_grad_en: 算法 grad 条件是否使能, 0: 不使能, 1: 使能。

flat_level_sel: 平坦区模板选择, 模板参考上面 filter 的模板, 取值范围[0-8]。

pattern_level_sel: 纹理区模板选择, 模板参考上面 filter 的模板, 取值范围[0-8]。

edge_level_sel: 边缘区模板选择, 模板参考上面 filter 的模板, 取值范围[0-8]。

3.2.11.6 MFD&UVNR

```
<MFD_Support index="1" type="double" size="[1 1]">
[0]
</MFD_Support>
<MFD_Gain index="1" type="double" size="[1 3]">
[2 4 6]
</MFD_Gain>
<MFD_Frames index="1" type="double" size="[1 3]">
[2 4 6]
</MFD_Frames>
<UVNR_Support index="1" type="double" size="[1 1]">
[0]
</UVNR_Support>
<UVNR_Gain index="1" type="double" size="[1 3]">
[2 4 8]
</UVNR_Gain>
<UVNR_Ratio index="1" type="double" size="[1 3]">
[2 4 6]
</UVNR_Ratio>
<UVNR_Distance index="1" type="double" size="[1 3]">
[2 4 6]
</UVNR_Distance>
..
```

此模块仅 hal00 代码可支持此功能，其他代码不支持。

MFD_Support: 是否使能多帧降噪，0: 不使能，1: 使能。

MFD_Gain: 不同 gain 下，控制使用不同的叠加去噪帧数。

MFD_Frames: 对应上面不同 gain，使用不同的叠加去噪帧数。

UVNR_Support: 是否使能 uv 双边滤波，0: 不使能，1: 使能。

UVNR_Gain: 不同 gain 下，控制使用不同的双边滤波半径和像素相似性的阈值。

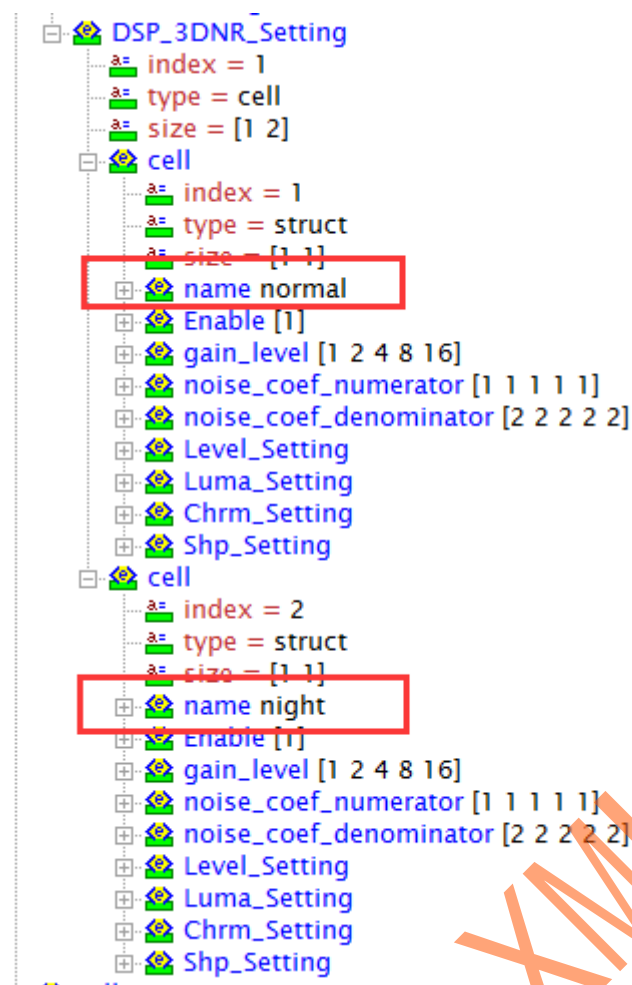
UVNR_Ratio: 对应上面不同 gain，使用不同的双边去噪半径。没有上限，推荐[0 - 20]。

UVNR_Distance: 对应上面不同 gain，使用不同的像素相似性阈值，主要关系到图像边缘区域。没有上限，推荐[0 - 10]。

3.2.11.7 DSP_3DNR_Setting

3Dnr 去噪参数设置仅 camera_engine_cifisp 代码且为 rv1108 带 DSP 芯片支持此功能目前也是每个分辨率下支持 2 种模式设置，normal 和 night 模式。该模块对外释放版本包含 2 个版本，

第一版 3dnr 参数:



name: 模式。目前只支持 2 种，normal 和 night 模式。

Enable: 是否打开 3dnr 功能。

gain_level: 3dnr 参数会随着 gain 值变化而变化。

noise_coef_numerator: 3dnr 噪声估计值调整参数，调整值中分子部分。

noise_coef_denominator: 3dnr 噪声估计值调整参数，调整值中分母部分。

3dnr 会对当前噪声进行估计，调整参数，是对此估计值进行乘以上面系数。

Level_Setting: 不同等级 3dnr 设置

```
<Level_Setting index="1" type="struct" size="[1 1]">
  <luma_sp_nr_en index="1" type="unsigned char" size="[1 1]">[1]</luma_sp_nr_en>
  <luma_sp_nr_level index="1" type="unsigned char" size="[1 5]">[3 6 9 12 16]</luma_sp_nr_level>
  <luma_te_nr_en index="1" type="unsigned char" size="[1 1]">[1]</luma_te_nr_en>
  <luma_te_nr_level index="1" type="unsigned char" size="[1 5]">[8 8 12 12 16]</luma_te_nr_level>
  <chrm_sp_nr_en index="1" type="unsigned char" size="[1 1]">[1]</chrm_sp_nr_en>
  <chrm_sp_nr_level index="1" type="unsigned char" size="[1 5]">[16 16 16 16 16]</chrm_sp_nr_level>
  <chrm_te_nr_en index="1" type="unsigned char" size="[1 1]">[1]</chrm_te_nr_en>
  <chrm_te_nr_level index="1" type="unsigned char" size="[1 5]">[16 16 16 16 16]</chrm_te_nr_level>
  <shp_en index="1" type="unsigned char" size="[1 1]">[1]</shp_en>
  <shp_level index="1" type="unsigned char" size="[1 5]">[16 16 16 16 16]</shp_level>
</Level_Setting>
```

luma_sp_nr_en: 空间域亮度去噪功能是否打开，0: 关闭， 1: 打开

luma_sp_nr_level: 空间域亮度去噪配置不同 gain 下采用去噪力度等级。

luma_te_nr_en: te 亮度去噪功能是否打开，0: 关闭， 1: 打开

luma_te_nr_level: te 亮度去噪配置不同 gain 下采用去噪力度等级。

chrom_sp_nr_en: sp 色度去噪功能是否打开, 0: 关闭, 1: 打开
chrom_sp_nr_level: sp 色度去噪配置不同 gain 下采用去噪力度等级。
chrom_te_nr_en: te 色度去噪功能是否打开, 0: 关闭, 1: 打开
chrom_te_nr_level: te 色度去噪配置不同 gain 下采用去噪力度等级。
shp_en: 亮度锐化功能是否打开, 0: 关闭, 1: 打开
shp_level: 亮度锐化配置不同 gain 下采用锐化力度等级。

Luma_Setting: 3dnr 亮度去噪配置

```
<Luma_Setting index="1" type="struct" size="[1 1]">
  <luma_default index="1" type="unsigned char" size="[1 1]">[1]</luma_default>
  <luma_sp_rad index="1" type="unsigned char" size="[1 5]">[3 3 3 3 3]</luma_sp_rad>
  <luma_te_max_bi_num index="1" type="unsigned char" size="[1 5]">[0 0 0 0 0]</luma_te_max_bi_num>
  <luma_w00 index="1" type="unsigned char" size="[1 5]">[2 2 2 2 2]</luma_w00>
  <luma_w01 index="1" type="unsigned char" size="[1 5]">[6 6 6 6 6]</luma_w01>
  <luma_w02 index="1" type="unsigned char" size="[1 5]">[12 12 12 12 12]</luma_w02>
  <luma_w03 index="1" type="unsigned char" size="[1 5]">[6 6 6 6 6]</luma_w03>
  <luma_w04 index="1" type="unsigned char" size="[1 5]">[2 2 2 2 2]</luma_w04>
  <luma_w10 index="1" type="unsigned char" size="[1 5]">[6 6 6 6 6]</luma_w10>
  <luma_w11 index="1" type="unsigned char" size="[1 5]">[30 30 30 30 30]</luma_w11>
  <luma_w12 index="1" type="unsigned char" size="[1 5]">[48 48 48 48 48]</luma_w12>
  <luma_w13 index="1" type="unsigned char" size="[1 5]">[30 30 30 30 30]</luma_w13>
  <luma_w14 index="1" type="unsigned char" size="[1 5]">[6 6 6 6 6]</luma_w14>
  <luma_w20 index="1" type="unsigned char" size="[1 5]">[12 12 12 12 12]</luma_w20>
  <luma_w21 index="1" type="unsigned char" size="[1 5]">[48 48 48 48 48]</luma_w21>
  <luma_w22 index="1" type="unsigned char" size="[1 5]">[96 96 96 96 96]</luma_w22>
  <luma_w23 index="1" type="unsigned char" size="[1 5]">[48 48 48 48 48]</luma_w23>
  <luma_w24 index="1" type="unsigned char" size="[1 5]">[12 12 12 12 12]</luma_w24>
  <luma_w30 index="1" type="unsigned char" size="[1 5]">[6 6 6 6 6]</luma_w30>
  <luma_w31 index="1" type="unsigned char" size="[1 5]">[30 30 30 30 30]</luma_w31>
  <luma_w32 index="1" type="unsigned char" size="[1 5]">[48 48 48 48 48]</luma_w32>
  <luma_w33 index="1" type="unsigned char" size="[1 5]">[30 30 30 30 30]</luma_w33>
  <luma_w34 index="1" type="unsigned char" size="[1 5]">[6 6 6 6 6]</luma_w34>
  <luma_w40 index="1" type="unsigned char" size="[1 5]">[2 2 2 2 2]</luma_w40>
  <luma_w41 index="1" type="unsigned char" size="[1 5]">[6 6 6 6 6]</luma_w41>
  <luma_w42 index="1" type="unsigned char" size="[1 5]">[12 12 12 12 12]</luma_w42>
  <luma_w43 index="1" type="unsigned char" size="[1 5]">[6 6 6 6 6]</luma_w43>
  <luma_w44 index="1" type="unsigned char" size="[1 5]">[2 2 2 2 2]</luma_w44>
</Luma_Setting>
```

luma_default: 1: 采用代码默认配置, 0: 采用底下参数配置。

luma_sp_rad: 亮度 Y 通道对应空间双边滤波器模板大小。

luma_te_max_bi_num: 暂时未用。

luma_w00 - luma_w44: 随着 gain 不同, 配置亮度去噪 5x5 模板权重。

最中心一行权重为 8bit 取值范围, 其余权重取值范围为 6bit, 不带符号位。

Chrm_Setting: 3dnr 色度去噪配置

```
<Chrm_Setting index="1" type="struct" size="[1 1]">
  <chrm_default index="1" type="unsigned char" size="[1 1]">[1]</chrm_default>
  <chrm_sp_rad index="1" type="unsigned char" size="[1 5]">[3 3 3 3 3]</chrm_sp_rad>
  <chrm_te_max_bi_num index="1" type="unsigned char" size="[1 5]">[0 0 0 0 0]</chrm_te_max_bi_num>
  <chrm_w00 index="1" type="unsigned char" size="[1 5]">[2 2 2 2 2]</chrm_w00>
  <chrm_w01 index="1" type="unsigned char" size="[1 5]">[5 5 5 5 5]</chrm_w01>
  <chrm_w02 index="1" type="unsigned char" size="[1 5]">[5 5 5 5 5]</chrm_w02>
  <chrm_w03 index="1" type="unsigned char" size="[1 5]">[5 5 5 5 5]</chrm_w03>
  <chrm_w04 index="1" type="unsigned char" size="[1 5]">[2 2 2 2 2]</chrm_w04>
  <chrm_w10 index="1" type="unsigned char" size="[1 5]">[5 5 5 5 5]</chrm_w10>
  <chrm_w11 index="1" type="unsigned char" size="[1 5]">[18 18 18 18 18]</chrm_w11>
  <chrm_w12 index="1" type="unsigned char" size="[1 5]">[25 25 25 25 25]</chrm_w12>
  <chrm_w13 index="1" type="unsigned char" size="[1 5]">[18 18 18 18 18]</chrm_w13>
  <chrm_w14 index="1" type="unsigned char" size="[1 5]">[5 5 5 5 5]</chrm_w14>
  <chrm_w20 index="1" type="unsigned char" size="[1 5]">[5 5 5 5 5]</chrm_w20>
  <chrm_w21 index="1" type="unsigned char" size="[1 5]">[25 25 25 25 25]</chrm_w21>
  <chrm_w22 index="1" type="unsigned char" size="[1 5]">[80 80 80 80 80]</chrm_w22>
  <chrm_w23 index="1" type="unsigned char" size="[1 5]">[25 25 25 25 25]</chrm_w23>
  <chrm_w24 index="1" type="unsigned char" size="[1 5]">[5 5 5 5 5]</chrm_w24>
  <chrm_w30 index="1" type="unsigned char" size="[1 5]">[5 5 5 5 5]</chrm_w30>
  <chrm_w31 index="1" type="unsigned char" size="[1 5]">[18 18 18 18 18]</chrm_w31>
  <chrm_w32 index="1" type="unsigned char" size="[1 5]">[25 25 25 25 25]</chrm_w32>
  <chrm_w33 index="1" type="unsigned char" size="[1 5]">[18 18 18 18 18]</chrm_w33>
  <chrm_w34 index="1" type="unsigned char" size="[1 5]">[5 5 5 5 5]</chrm_w34>
  <chrm_w40 index="1" type="unsigned char" size="[1 5]">[2 2 2 2 2]</chrm_w40>
  <chrm_w41 index="1" type="unsigned char" size="[1 5]">[5 5 5 5 5]</chrm_w41>
  <chrm_w42 index="1" type="unsigned char" size="[1 5]">[5 5 5 5 5]</chrm_w42>
  <chrm_w43 index="1" type="unsigned char" size="[1 5]">[5 5 5 5 5]</chrm_w43>
  <chrm_w44 index="1" type="unsigned char" size="[1 5]">[2 2 2 2 2]</chrm_w44>
</Chrm_Setting>
```

chrm_default: 1: 采用代码默认配置, 0: 采用底下参数配置。

chrm_sp_rad: 色度 cr cb 通道对应的空间双边滤波器模板大小。

chrm_te_max_bi_num: 暂时未用。

chrm_w00 ——chrm_w44: 随着 gain 不同, 配置色度去噪 5x5 模板权重。

最中心一行权重为 8bit 取值范围, 其余权重取值范围为 6bit, 不带符号位。

Shp_Setting: 3dnr 中亮度锐化参数配置

```
<Shp_Setting index="1" type="struct" size="[1 1]">
  <shp_default index="1" type="unsigned char" size="[1 1]">[1]</shp_default>
  <src_shp_thr index="1" type="double" size="[1 5]">[15 15 15 15 15]</src_shp_thr>
  <src_shp_div index="1" type="double" size="[1 5]">[5 5 5 5 5]</src_shp_div>
  <src_shp_l index="1" type="double" size="[1 5]">[1 1 1 1 1]</src_shp_l>
  <src_shp_c index="1" type="double" size="[1 5]">[0 0 0 0 0]</src_shp_c>
  <src_shp_w00 index="1" type="double" size="[1 5]">[-1 -1 -1 -1 -1]</src_shp_w00>
  <src_shp_w01 index="1" type="double" size="[1 5]">[-1 -1 -1 -1 -1]</src_shp_w01>
  <src_shp_w02 index="1" type="double" size="[1 5]">[-2 -2 -2 -2 -2]</src_shp_w02>
  <src_shp_w03 index="1" type="double" size="[1 5]">[-1 -1 -1 -1 -1]</src_shp_w03>
  <src_shp_w04 index="1" type="double" size="[1 5]">[-1 -1 -1 -1 -1]</src_shp_w04>
  <src_shp_w10 index="1" type="double" size="[1 5]">[-1 -1 -1 -1 -1]</src_shp_w10>
  <src_shp_w11 index="1" type="double" size="[1 5]">[2 2 2 2 2]</src_shp_w11>
  <src_shp_w12 index="1" type="double" size="[1 5]">[2 2 2 2 2]</src_shp_w12>
  <src_shp_w13 index="1" type="double" size="[1 5]">[2 2 2 2 2]</src_shp_w13>
  <src_shp_w14 index="1" type="double" size="[1 5]">[-1 -1 -1 -1 -1]</src_shp_w14>
  <src_shp_w20 index="1" type="double" size="[1 5]">[-2 -2 -2 -2 -2]</src_shp_w20>
  <src_shp_w21 index="1" type="double" size="[1 5]">[2 2 2 2 2]</src_shp_w21>
  <src_shp_w22 index="1" type="double" size="[1 5]">[36 36 36 36 36]</src_shp_w22>
  <src_shp_w23 index="1" type="double" size="[1 5]">[2 2 2 2 2]</src_shp_w23>
  <src_shp_w24 index="1" type="double" size="[1 5]">[-2 -2 -2 -2 -2]</src_shp_w24>
  <src_shp_w30 index="1" type="double" size="[1 5]">[-1 -1 -1 -1 -1]</src_shp_w30>
  <src_shp_w31 index="1" type="double" size="[1 5]">[2 2 2 2 2]</src_shp_w31>
  <src_shp_w32 index="1" type="double" size="[1 5]">[2 2 2 2 2]</src_shp_w32>
  <src_shp_w33 index="1" type="double" size="[1 5]">[2 2 2 2 2]</src_shp_w33>
  <src_shp_w34 index="1" type="double" size="[1 5]">[-1 -1 -1 -1 -1]</src_shp_w34>
  <src_shp_w40 index="1" type="double" size="[1 5]">[-1 -1 -1 -1 -1]</src_shp_w40>
  <src_shp_w41 index="1" type="double" size="[1 5]">[-1 -1 -1 -1 -1]</src_shp_w41>
  <src_shp_w42 index="1" type="double" size="[1 5]">[-2 -2 -2 -2 -2]</src_shp_w42>
  <src_shp_w43 index="1" type="double" size="[1 5]">[-1 -1 -1 -1 -1]</src_shp_w43>
  <src_shp_w44 index="1" type="double" size="[1 5]">[-1 -1 -1 -1 -1]</src_shp_w44>
</Shp_Setting>
```

shp_default: 1: 采用代码默认配置, 0: 采用底下参数配置。

src_shp_thr: 锐化阈值, 取值范围从[0 31]。

src_shp_div: 权重位移位数, 取值范围[0 7]。

src_shp_l: 亮度 y 通道锐化使能标志位 0: 不使能 1: 使能。

src_shp_c: 色度 cr cb 通道锐化使能标志位 0: 不使能 1: 使能。

src_shp_w00 —— src_shp_w44: 随着 gain 不同, 配置亮度锐化 5x5 模板权重。
最中心一行权重为 8bit 取值范围, 其余权重取值范围为 6bit, 带符号位。

第二版 3dnr 参数:

```
<NEW_DSP_3DNR_Setting index="1" type="cell" size="[1 1]">
  <cell index="1" type="struct" size="[1 1]">
    <name index="1" type="char" size="[1 6]">normal</name>
    <Enable index="1" type="double" size="[1 1]">[0]</Enable>
    <dpc_enable index="1" type="double" size="[1 1]">[1]</dpc_enable>
    <gain_level index="1" type="double" size="[1 5]">[1 2 4 8 16]</gain_level>
    <ynr_Setting index="1" type="struct" size="[1 1]">
      <ynr_enable index="1" type="double" size="[1 1]">[1]</ynr_enable>
      <tnr_enable index="1" type="double" size="[1 1]">[1]</tnr_enable>
      <iir_enable index="1" type="double" size="[1 1]">[1]</iir_enable>
      <ynr_level index="1" type="double" size="[1 5]">[1 2 3 4 5]</ynr_level>
    </ynr_Setting>
    <uvnr_Setting index="1" type="struct" size="[1 1]">
      <uvnr_enable index="1" type="double" size="[1 1]">[1]</uvnr_enable>
      <uvnr_level index="1" type="double" size="[1 5]">[4 6 8 10 12]</uvnr_level>
    </uvnr_Setting>
    <sharp_Setting index="1" type="struct" size="[1 1]">
      <sharp_enable index="1" type="double" size="[1 1]">[1]</sharp_enable>
      <sharp_level index="1" type="double" size="[1 5]">[0 1 2 3 4]</sharp_level>
    </sharp_Setting>
  </cell>
```

Name: 模式名称 现在支持白天(normal)和夜晚(night)两个模式。

Enable: 3dnr 整个模块开关。0: 关闭, 1: 打开

dpc_enable: 3dnr 中去坏点模块, 一些色彩经过 3dnr 可能变成坏点。0: 关闭, 1: 打开。

gain_level: 3dnr 的力度随着 gain 值变化而变化。这个是在不同 gain 下, 对应下面不同设置

ynr_Setting: 3dnr 针对亮度去噪的设置。

ynr_enable: 亮度噪声空域去噪使能参数, 0: 关闭, 1: 打开, 默认值 1。

tnr_enable: 亮度噪声时域去噪使能参数, 0: 关闭, 1: 打开, 默认值 1。

ynr_enable=0, 默认不使能。

iir_enable: 亮度噪声某个去噪功能使能参数, 0: 关闭, 1: 打开, 默认值 0。

ynr_enable=0, 默认不使能。

ynr_level: 亮度去噪力度, 随着上面 gain 变化而变化。

去噪力度等级[1 - 5], 默认值 3。

uvnr_Setting: 针对彩色噪声的去噪设置

uvnr_enable: 彩噪去除使能参数, 0: 关闭, 1: 打开, 默认值 1。

uvnr_level: 彩噪去噪力度, 随着上面 gain 变化而变化。

去噪力度等级[4 - 16], 默认值 12。

sharp_Setting: 锐化相关设置

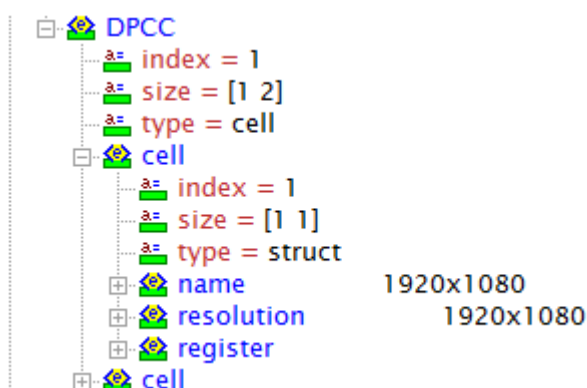
sharp_enable: 锐化模块使能参数, 0: 关闭, 1: 打开, 默认值 1。

去噪力度等级[0 - 4], 默认值 2。

3.2.12 DPCC 参数说明

DPCC 模块全称: Defect Pixel Cluster Correction。

DPCC 去坏点参数, 此模块为设置相应寄存器为相应的值。



此模块参数均为相应寄存器对应寄存器值设置。

3.2.12.1 dpcc IQ 参数说明

ISP_DPCC_MODE:

Bits	Name	Description
31:3	---	unused
2	STAGE1_ENABLE	1: enable stage1 *Default* 0: bypass stage1

Bits	Name	Description
1	GRAYSCALE_MODE	1: enable gray scale data input from black and white sensors (without color filter array) 0: BAYER DATA INPUT *Default*
0	ISP_DPCC_enable	1: enable DPCC 0: bypass DPCC *Default*

ISP_DPCC_OUT_MODE:

Bits	Name	Description
31:4	---	unused
3	STAGE1_RB_3x3	1: stage1 red/blue 9 pixel (3x3) output median 0: stage1 red/blue 4 or 5 pixel output median *Default*
2	STAGE1_G_3x3	1: stage1 green 9 pixel (3x3) output median 0: stage1 green 4 or 5 pixel output median *Default*
1	STAGE1_INCL_RB_CENTER	1: stage1 include center pixel for red/blue output median 2x2+1 *Default* 0: stage1 do not include center pixel for red/blue output median 2x2
0	STAGE1_INCL_GREEN_CENTER	1: stage1 include center pixel for green output median 2x2+1 *Default* 0: stage1 do not include center pixel for green output median 2x2

ISP_DPCC_SET_USE:

Bits	Name	Description
31:4	---	unused
3	STAGE1_USE_FIX_SET	1: stage1 use hard coded methods set *Default* 0: stage1 do not use hard coded methods set

Bits	Name	Description
2	STAGE1_USE_SET_3	1: stage1 use methods set 3 0: stage1 do not use methods set 3 *Default*
1	STAGE1_USE_SET_2	1: stage1 use methods set 2 0: stage1 do not use methods set 2 *Default*
0	STAGE1_USE_SET_1	1: stage1 use methods set 1 *Default* 0: stage1 do not use methods set 1
Note: methods sets can be used in parallel for each stage and the result is the logical OR of all selected sets		

ISP_DPCC_METHODS_SET1:

Bits	Name	Description
31:13	---	unused
12	RG_RED_BLUE1_ENABLE	1: enable Rank Gradient check for red_blue *Default* 0: bypass Rank Gradient check for red_blue
11	RND_RED_BLUE1_ENABLE	1: enable Rank Neighbor Difference check for red_blue *Default* 0: bypass Rank Neighbor Difference check for red_blue
10	RO_RED_BLUE1_ENABLE	1: enable Rank Order check for red_blue *Default* 0: bypass Rank Order check for red_blue
9	LC_RED_BLUE1_ENABLE	1: enable Line check for red_blue *Default* 0: bypass Line check for red_blue
8	PG_RED_BLUE1_ENABLE	1: enable Peak Gradient check for red_blue *Default* 0: bypass Peak Gradient check for red_blue
7:5	---	unused
4	RG_GREEN1_ENABLE	1: enable Rank Gradient check for green *Default* 0: bypass Rank Gradient check for green
3	RND_GREEN1_ENABLE	1: enable Rank Neighbor Difference check for green *Default* 0: bypass Rank Neighbor Difference check for green
2	RO_GREEN1_ENABLE	1: enable Rank Order check for green *Default* 0: bypass Rank Order check for green
Note: different methods can be used in parallel, the result is the logical AND of all selected methods		

ISP_DPCC_METHODS_SET2: 同上 set1

ISP_DPCC_METHODS_SET3: 同上 set1

SET1 对应阈值参数:

ISP_DPCC_LINE_THRESH_1:

Bits	Name	Description
31:16	---	unused
15:8	LINE_THR_1_RB	line threshold for set 1 red/blue
7:0	LINE_THR_1_G	line threshold for set 1 green
Note: all values are unsigned integer		

ISP_DPCC_LINE_MAD_FAC_1:

Bits	Name	Description
31:14	---	unused
13:8	LINE_MAD_FAC_1_RB	line MAD factor for set 1 red/blue
7:6	---	unused
5:0	LINE_MAD_FAC_1_G	line MAD factor for set 1 green
Note: all values are unsigned integer		

ISP_DPCC_PG_FAC_1:

Bits	Name	Description
31:14	---	unused
13:8	PG_FAC_1_RB	Peak gradient factor for set 1 red/blue
7:6	---	unused
5:0	PG_FAC_1_G	Peak gradient factor for set 1 green
Note: all values are unsigned integer		

ISP_DPCC_RND_THRESH_1:

Bits	Name	Description
31:16	---	unused
15:8	RND_THR_1_RB	Rank Neighbor Difference threshold for set 1 red/blue
7:0	RND_THR_1_G	Rank Neighbor Difference threshold for set 1 green
Note: all values are unsigned integer		

ISP_DPCC_RG_FAC_1:

Bits	Name	Description
31:14	---	unused
13:8	RG_FAC_1_RB	Rank gradient factor for set 1 red/blue
7:6	---	unused
5:0	RG_FAC_1_G	Rank gradient factor for set 1 green
Note: all values are unsigned integer		

ISP_DPCC_RO_LIMITS:

Bits	Name	Description
31:12	---	unused
11:10	RO_LIM_3_RB	Rank order limit for set 3 red/blue
9:8	RO_LIM_3_G	Rank order limit for set 3 green
7:6	RO_LIM_2_RB	Rank order limit for set 2 red/blue
5:4	RO_LIM_2_G	Rank order limit for set 2 green
3:2	RO_LIM_1_RB	Rank order limit for set 1 red/blue
1:0	RO_LIM_1_G	Rank order limit for set 1 green
Note: all values are unsigned integer		

ISP_DPCC_RND_OFFS:

Bits	Name	Description
31:12	---	unused
11:10	RND_OFFS_3_RB	Rank Offset to Neighbor for set 3 red/blue
9:8	RND_OFFS_3_G	Rank Offset to Neighbor for set 3 green

Bits	Name	Description
7:6	RND_OFFS_2_RB	Rank Offset to Neighbor for set 2 red/blue
5:4	RND_OFFS_2_G	Rank Offset to Neighbor for set 2 green
3:2	RND_OFFS_1_RB	Rank Offset to Neighbor for set 1 red/blue
1:0	RND_OFFS_1_G	Rank Offset to Neighbor for set 1 green

Note: all values are unsigned integer

剩余参数为 set2 和 set3 可调参数，与 set1 相同意思，在此不再复述。

3.2.12.3 dpcc 推荐设置

Datasheet 中推荐 6 组参数设置，根据不同坏点数量和坏点大小进行设置，如下图：

Case	2. single pixels	3. small clusters	4. big clusters
a) few defects	par1	par2	par3
b) many defects	par4	par5	par6

register	par1	par2	par3	par4	par5	par6
ISP_DPCC_MODE	0x0005	0x0005	0x0005	0x0005	0x0005	0x0005
ISP_DPCC_OUT_MODE	0x0003	0x0003	0x0003	0x0003	0x0003	0x0003
ISP_DPCC_SET_USE	0x0003	0x0003	0x0007	0x0003	0x000F	0x000F
ISP_DPCC_METHODS_SET1	0x1F1F	0x1F1F	0x1D1D	0x1D1D	0x1D1D	0x1D1D
ISP_DPCC_METHODS_SET2	0x0707	0x0707	0x0707	0x0707	0x0707	0x0707
ISP_DPCC_METHODS_SET3	0x1F1F	0x1F1F	0x1F1F	0x1F1F	0x1F1F	0x1F1F
ISP_DPCC_LINE_THRESH_1	0x0808	0x0808	0x0808	0x0808	0x0808	0x0808
ISP_DPCC_LINE_MAD_FAC_1	0x0404	0x0404	0x0404	0x0404	0x0404	0x0404
ISP_DPCC_PG_FAC_1	0x0808	0x0808	0x0806	0x0806	0x0606	0x0404
ISP_DPCC_RND_THRESH_1	0x0A0A	0x0A0A	0x0A0A	0x0A0A	0x0808	0x0804
ISP_DPCC_RG_FAC_1	0x2020	0x2020	0x2020	0x2020	0x1010	0x0802
ISP_DPCC_LINE_THRESH_2	0x2018	0x1010	0x100C	0x100C	0x100C	0x100C
ISP_DPCC_LINE_MAD_FAC_2	0x1810	0x1810	0x1810	0x1810	0x0808	0x0404
ISP_DPCC_PG_FAC_2	0x0806	0x0806	0x0806	0x0806	0x0606	0x0404
ISP_DPCC_RND_THRESH_2	0x0808	0x0808	0x0808	0x0808	0x0808	0x0808
ISP_DPCC_RG_FAC_2	0x0808	0x0808	0x0808	0x0808	0x0808	0x0808
ISP_DPCC_LINE_THRESH_3	0x2020	0x2020	0x2020	0x2020	0x0808	0x0000
ISP_DPCC_LINE_MAD_FAC_3	0x0404	0x0404	0x0404	0x0404	0x0404	0x0404
ISP_DPCC_PG_FAC_3	0x0A0A	0x0A0A	0x0A0A	0x0A0A	0x0606	0x0404
ISP_DPCC_RND_THRESH_3	0x0806	0x0806	0x0806	0x0806	0x0806	0x0804
ISP_DPCC_RG_FAC_3	0x0404	0x0404	0x0404	0x0404	0x0404	0x0400

ISP_DPCC_RO_LIMITS	0x09A5	0x09FA	0x0AFA	0x09FA	0x0FFF	0x0FFF
ISP_DPCC_RND_OFFS	0x0AAA	0x0FFF	0x0FFF	0x0FFF	0x0FFF	0x0FFF

3.2.12.4 isp12 修改 dpcc

ISP12 芯片对坏点检测算法进行了一点更新，如果要使用旧 dpcc 算法，需将 dpcc_mode 寄存器第 4bit 置为 1，如果第 4bit 为 0，则使用带 rk mode 条件的新 dpcc 算法。

寄存器上，isp12 的 dpcc 寄存器大部分和旧模式相同。少量 rk method 新增变量是在原来寄存器上空余位增加对应的变量。

如下图：

isp_dpcc_2900 ISP_DPCC_MODE

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4	RW	0x1	sw_mem_update_mode 1'b0: select uncorrected data to update mem(rk mode) 1'b1: select updated data to update mem(img mode default)

isp_dpcc_2900 ISP_DPCC_OUTPUT_MODE

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6:5	RW	0x0	sw_rk_out_sel 2'b00:RK method1 2'b01:RK method2 2'b10:RK method3 2'b11:reserved
4	RW	0x0	sw_dpcc_output_sel 1'b0: select median mode 1'b1: select rk output mode

isp_dpcc_2900_ISP_DPCC_METHODS_SET_1

Address: Operational Base + offset (0x000c)

Bit	Attr	Reset Value	Description
31:14	RO	0x0	reserved

Copyright 2015 @FuZhou Rockchip Electronics Co., Ltd.
298

PX30 TRM

13	RW	0x0	sw_rk_red_blue1_en 1: enable RK method check for green *Default* 0: bypass RK method check for green
5	RW	0x0	sw_rk_green1_en 1: enable RK method check for green *Default* 0: bypass RK method check for green

isp_dpcc_2900_ISP_DPCC_LINE_THRESH_1

Address: Operational Base + offset (0x0018)

Bit	Attr	Reset Value	Description
31:24	RW	0x00	sw_mindis1_rb min distance for set 1 red /blue

Copyright 2015 @FuZhou Rockchip Electronics Co., Ltd.
303

PX30 TRM

23:16	RW	0x00	sw_mindis1_g min distance for set 1 green
-------	----	------	---

isp_dpcc_2900_ISP_DPCC_LINE_MAD_FAC_1

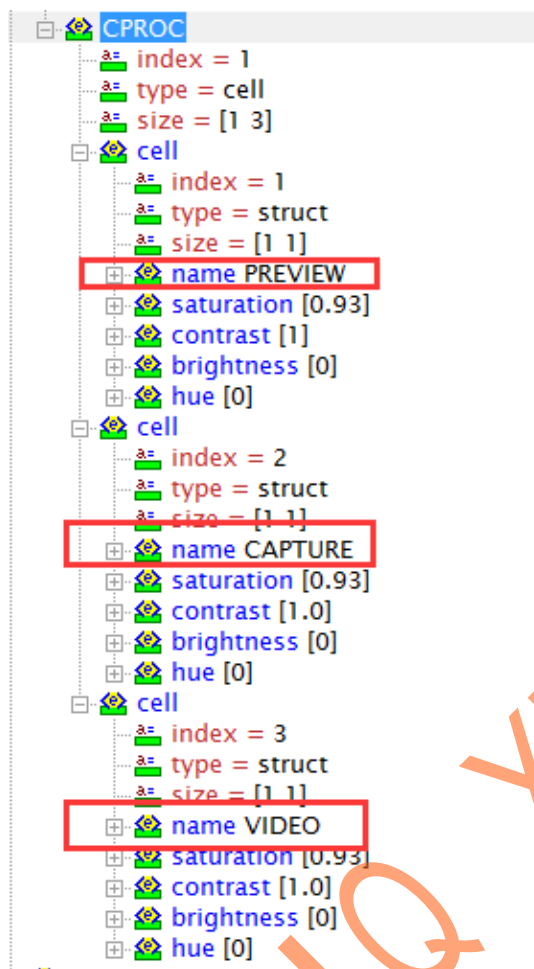
Address: Operational Base + offset (0x001c)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:24	RW	0x00	sw_dis_scale_min1
23:22	RO	0x0	reserved
21:16	RW	0x00	sw_dis_scale_max1

3.2.13 CPROC 参数说明

CPROC 模块全称: color process

Cproc 设置图像明度, 对比度, 饱和度, 色调参数



目前支持 3 个模式的设置, 分别为 preview, capture, video 模式

Name: 模式名称

Saturation: 饱和度设置, 取值范围[0-1.992]

Contrast: 对比度设置, 取值范围[0-1.992]

Brightness: 明度设置, 取值范围[-255 +255]

Hue: 色调设置, 取值范围[-90 87.188]

3.2.14 IESHARPEN 参数说明



此模块目前只有 ISP12 芯片上的 isp 才具有的功能。

IESharpenEnable: IE 模块的 rk 设计锐化是否使能：0：不使能，1：使能。

coring_thr: y 通道拉普拉斯锐化结果，如果小于此阈值，则不锐化。

full_range: yuv 是否采用 full range, 0:不是，1：是。

switch_avg:

中心点与周围 3x3 均值的绝对差值比较条件是否使能，0：不使能，1：使能。

yavg_thr:

判断中心点周围 3x3 均值，用 4 个阈值划分出 5 个区域，对每个区域采用不同参数进行锐化控制。取值范围[0-255]。

p_delta1:

上面所说 5 个区域中，每个区域对应中心点和周围 3x3 均值的绝对差值的阈值设定，用来判断是否是平坦区和噪声区的条件之一。取值范围[0-255]。

p_delta2:

上面所说 5 个区域中，每个区域周围 3x3 像素点和均值的绝对差值的阈值设定，用来判断是否是平坦区和噪声区的条件之二。取值范围[0-255]。

Pmaxnumber:

上面所说 5 个区域中,每个区域周围 3x3 像素点与均值绝对差值小于 p_delta2 的最少个数的阈值设定,用来判断是否是平坦区的条件之三。取值范围[0-8]。

Pminnumber:

上面所说 5 个区域中,每个区域周围 3x3 像素点与均值绝对差值小于 p_delta2 的最多个数的阈值设定,用来判断是否是噪声区的条件之三。取值范围[0-8]。

gauss_flat_coe:

平坦区 y 通道对应的 3x3 高斯滤波系数。取值范围[0-255]。

gauss_noise_coe:

噪声区 y 通道对应的 3x3 高斯滤波系数。取值范围[0-255]。

gauss_other_coe:

其他区 y 通道对应的 3x3 高斯滤波系数。取值范围[0-255]。

低 gain 对应滤波参数和锐化参数: 1.0-4.0 此 gain 规定目前写死在驱动代码中

l_p_grad:

y 通道 3x3 像素点进行滤波后, h 和 v 方向总梯度差作为判断, 用 4 个阈值将梯度差范围划分出 5 个区域, 用来控制不同的锐化系数。取值范围[0-2047]。

l_sharp_factor: 上面 5 个梯度差区域, 对应的不同的锐化系数。取值范围[0-63]。

l_line1_filter_coe:

y 通道 3x3 像素点进行滤波, 第一行数据对应滤波系数。取值范围[0-63]。

l_line2_filter_coe:

y 通道 3x3 像素点进行滤波, 第二行数据对应滤波系数。取值范围[0-63]。

l_line3_filter_coe:

y 通道 3x3 像素点进行滤波, 第三行数据对应滤波系数。取值范围[0-63]。

高 gain 对应滤波参数和锐化参数: gain>4.0 此 gain 规定目前写死在驱动代码中

h_p_grad: 同上, 高 gain 对应系数。

h_sharp_factor: 同上, 高 gain 对应系数。

h_line1_filter_coe: 同上, 高 gain 对应系数。

h_line2_filter_coe: 同上, 高 gain 对应系数。

h_line3_filter_coe: 同上, 高 gain 对应系数。

uv_gauss_flat_coe: 平坦区 uv 通道对应的 3x5 高斯滤波系数。取值范围[0-255]。

uv_gauss_noise_coe: 噪声区 uv 通道对应的 3x5 高斯滤波系数。取值范围[0-255]。

uv_gauss_other_coe: 其他区 uv 通道对应的 3x5 高斯滤波系数。取值范围[0-255]。

3.3 system 参数说明

System 里的 afps 参数，目前软件上已按照曝光分解方式自动去变帧率了，这里保持原样即可。

