



Gowin ISP 系统 用户手册

UG954-1.0,2021-07-20

版权所有 © 2021 广东高云半导体科技股份有限公司

GOWIN高云, Gowin, 高云均为广东高云半导体科技股份有限公司注册商标, 本手册中提到的其他任何商标, 其所有权利属其拥有者所有。未经本公司书面许可, 任何单位和个人都不得擅自摘抄、复制、翻译本档内容的部分或全部, 并不得以任何形式传播。

免责声明

本文档并未授予任何知识产权的许可, 并未以明示或暗示, 或以禁止发言或其它方式授予任何知识产权许可。除高云半导体在其产品的销售条款和条件中声明的责任之外, 高云半导体概不承担任何法律或非法律责任。高云半导体对高云半导体产品的销售和 / 或使用不作任何明示或暗示的担保, 包括对产品的特定用途适用性、适销性或对任何专利权、版权或其它知识产权的侵权责任等, 均不作担保。高云半导体对文档中包含的文字、图片及其它内容的准确性和完整性不承担任何法律或非法律责任, 高云半导体保留修改文档中任何内容的权利, 恕不另行通知。高云半导体不承诺对这些文档进行适时的更新。

版本信息

日期	版本	说明
2021/07/20	1.0	初始版本。

目录

目录	i
图目录	ii
表目录	iii
1 关于本手册	1
1.1 手册内容	1
1.2 相关文档	1
1.3 技术支持与反馈	1
1.4 术语、缩略语	2
2 ISP 简述	3
2.1 结构	3
2.2 特性	4
2.3 资源统计	5
3 ISP 系统模块	6
3.1 CFA 模块	6
3.2 CCM 模块	7
3.3 GAMMA 模块	7
3.4 AEAWB 模块	8
3.5 CSC 模块	9
3.6 Scaler 模块	9
4 ISP 应用参考设计	10
4.1 Demo 设计采用的硬件平台	10
4.2 Demo 结构设计说明	10
4.3 MCU 指令控制	11

图目录

图 2-1 典型精简的 ISP 结构	3
图 2-2 ISP 系统设计	4
图 3-1 CFA 模块	6
图 3-2 CFA 模块接口	6
图 3-3 CCM 模块	7
图 3-4 CCM 模块接口	7
图 3-5 GAMMA 模块	7
图 3-6 GAMMA 模块接口	8
图 3-7 AEAWB 模块接口	8
图 3-8 CSC 模块接口	9
图 3-9 Scaler 模块接口	9
图 4-1 Demo 结构	10

表目录

表 1-1 术语、缩略语	2
表 2-1 ISP Demo 资源	5
表 2-2 ISP 资源	5
表 4-1 寄存器读写操作	12

1 关于本手册

1.1 手册内容

- ISP 简述
- ISP 模块介绍
- ISP 应用参考设计

1.2 相关文档

通过登录高云半导体网站 www.gowinsemi.com.cn 可以下载、查看以下相关文档：

- [IPUG782, Gowin AEAWB IP 用户指南](#)
- [IPUG757, Gowin Color Filter Array Interpolation IP 用户指南](#)
- [IPUG766, Gowin Color Correction Matrix IP 用户指南](#)
- [IPUG765, Gowin Gamma Correction IP 用户指南](#)
- [IPUG769, Gowin Video Frame Buffer IP 用户指南](#)
- [IPUG902, Gowin CSC IP 用户指南](#)
- [IPUG903, Gowin Scaler IP 用户指南](#)

1.3 技术支持与反馈

高云半导体提供全方位的技术支持，在使用过程中如有任何疑问或建议，可直接与公司联系：

网址：www.gowinsemi.com.cn

E-mail：support@gowinsemi.com

Tel: +86 755 8262 039

1.4 术语、缩略语

本手册中出现的相关术语、缩略语及相关释义如表 1-1 所示。

表 1-1 术语、缩略语

术语、缩略语	全称	含义
ISP	Image Signal Processing	图像信号处理
CFA	Color Filter Array Interpolation	色彩滤波阵列
CCM	Color Correction Matrix	颜色校正阵列
CSC	Color Space Conversion	颜色空间变换
BLC	BlackLevel Correction	黑平衡校正
DPC	Defect Pixel Correction	坏点校正
AE	Auto Exposure	自动曝光
AWB	Auto White Balance	自动白平衡

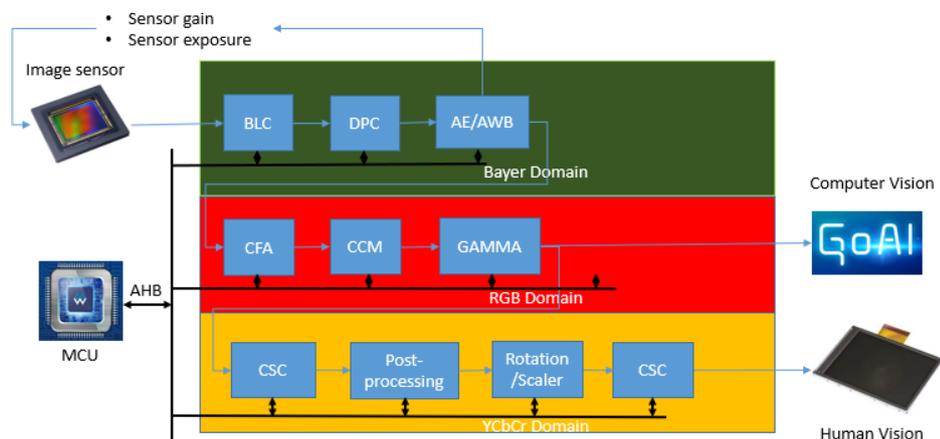
2 ISP 简述

2.1 结构

图像信号处理(ISP, Image Signal Processing), 是用来对前端图像传感器输出信号进行处理的单元。现实中, 由于镜头和传感器存在物理的缺陷, 以及传感器需要对不同光线环境进行适应, 所以需要 ISP 对传感器控制和图像信号处理, 从而得到预期的图像质量。

典型精简的 ISP 结构如图 2-1 所示。

图 2-1 典型精简的 ISP 结构

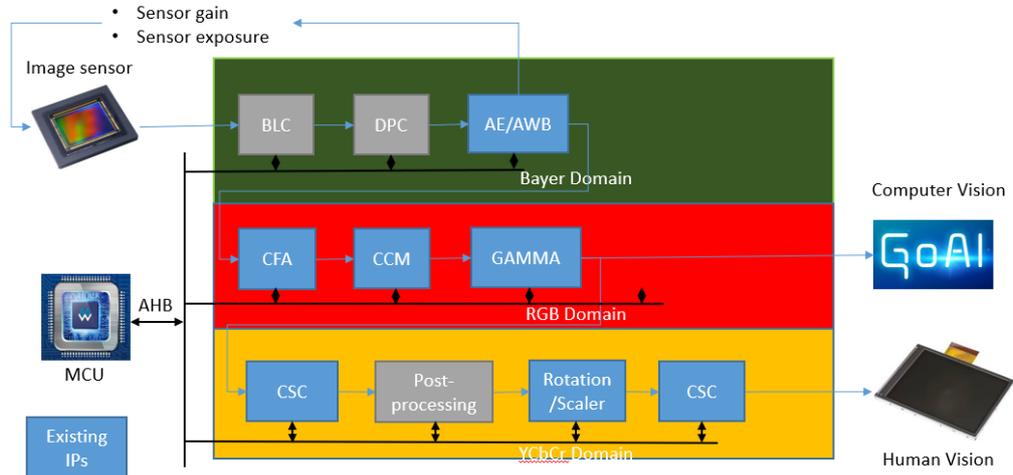


- BLC: 黑平衡校正模块。
- DPC: 坏点校正模块。
- AE: 自动曝光模块。
- AWB: 自动白平衡模块。
- CFA: 色彩滤波阵列模块。
- CCM: 颜色校正阵列模块。
- GAMMA: 伽马校正模块。

- CSC: 颜色空间变换模块。
- Scaler: 比例缩放模块。

目前, ISP 项目系统设计实现 ISP 主要模块(蓝色部分), 如图 2-2 所示。如果对灰色模块的设计有需要, 可以联系高云相关技术人员。

图 2-2 ISP 系统设计



2.2 特性

- 支持 4 种不同传感器对齐方式 (BGGR, GBRG, GRBG, RGGB), 以及图像数据位宽 8bit 和 10bit。
- 支持各种图像分辨率 (VGA, 720p, 1080p, 2K, 4K)。
- ISP 的模块都为独立的子模块, 包括,
 - AEAWB 模块的自动白平衡和自动曝光度调节。
 - CFA 模块将 Bayer 图像格式转为 RGB 图像格式。
 - CCM 模块实现颜色校正。
 - GAMMA 模块实现对图像灰度值的非线性调节。
 - CSC 模块实现颜色空间的转换。
 - Scaler 模块实现图像的缩放。

2.3 资源统计

ISP Demo 资源如表 2-1 所示。ISP 资源如表 2-2 所示。

表 2-1 ISP Demo 资源

模块名称	Register	LUT	ALU	DSP	BSRAM	SSRAM	PLL
Ov5647 Initialization	225	462	31	0	0	0	0
MIPI CSI-2 Interface	448	444	41	0	1	17	0
FOV Cropping	42	60	0	0	0	0	0
ISP Pipeline	2722	2013	1441	12	10	16	0
Video Frame Buffer	346	684	86	0	8	0	0
DDR3 Memory Interface	1729	1399	61	0	8	70	0
Scaler	892	623	68	0	18	4	0
Other	119	119	4	0	0	0	3
M1	2490	5694	162	3	64	20	0
Total	9165	11636	1894	13	109	131	3

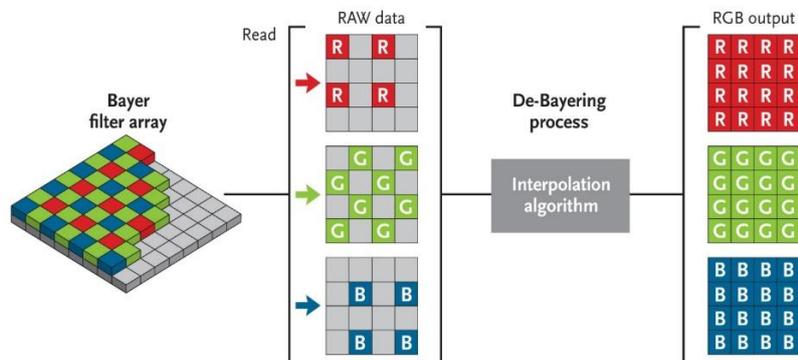
表 2-2 ISP 资源

模块名称	Register	LUT	ALU	DSP	BSRAM	SSRAM
CFA	597	570	693	0	6	0
CCM	578	56	0	6	0	0
GAMMA	76	22	0	0	1	0
AEAWB	1238	1153	748	6	3	4

3 ISP 系统模块

3.1 CFA 模块

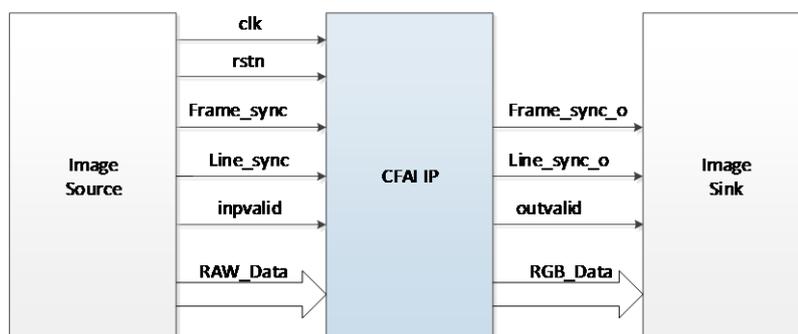
图 3-1 CFA 模块



Gowin Color Filter Array Interpolation IP 针对不同的传感器阵列排列模式，根据插值滤波器系数特点做了相应的优化，并且使用了 5×5 的高质量算子。根据缺失值的位置不同，在 R、G、B 插值中使用了不同的滤波器系数。

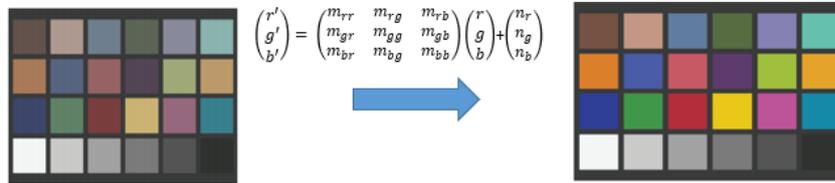
其模块主要接口如图 3-2 所示。

图 3-2 CFA 模块接口



3.2 CCM 模块

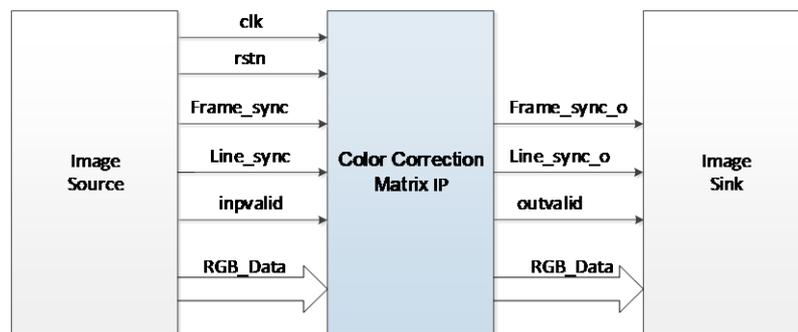
图 3-3 CCM 模块



CCM, Color Correction Matrix, 即色彩校正矩阵。色彩校正主要为了校正在滤光板处各颜色块之间的颜色渗透带来的颜色误差, 使映射后的颜色更符合人的认知习惯。色彩校正首先利用该图像传感器拍摄到的图像与标准图像相比较, 以此来计算得到一个校正矩阵。该矩阵就是该图像传感器的色彩校正矩阵 (CCM)。在该图像传感器应用的过程中, 便可以利用该矩阵对该图像传感器所拍摄的所有图像来进行校正, 以获得最接近于物体真实颜色的图像。

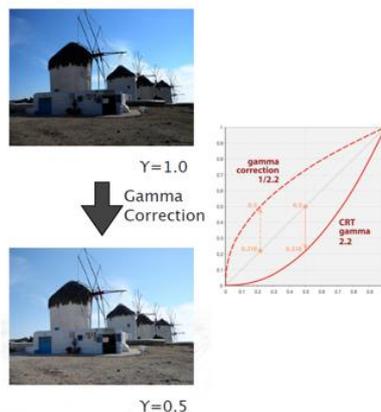
模块主要接口如图 3-4 所示。

图 3-4 CCM 模块接口



3.3 GAMMA 模块

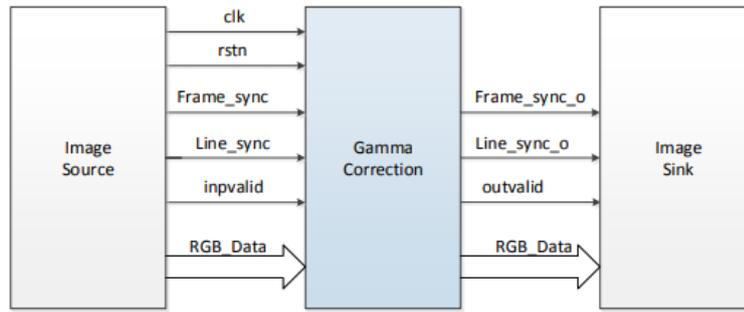
图 3-5 GAMMA 模块



Gamma Correction，即伽马校正，是一种对输入图像灰度值进行指数变换，进而校正亮度偏差的算法。人眼对于外部光源的感光值和输入光强呈指数型关系的。即在低照度下，人眼更容易分辨出亮度的变化，随着照度的增加，人眼不易分辨出亮度的变化。而摄像机感光与输入光强通常呈线性关系，为方便人眼辨识图像，需要将摄像机采集的图像进行 **gamma** 矫正。

其模块主要接口如图 3-6 所示。

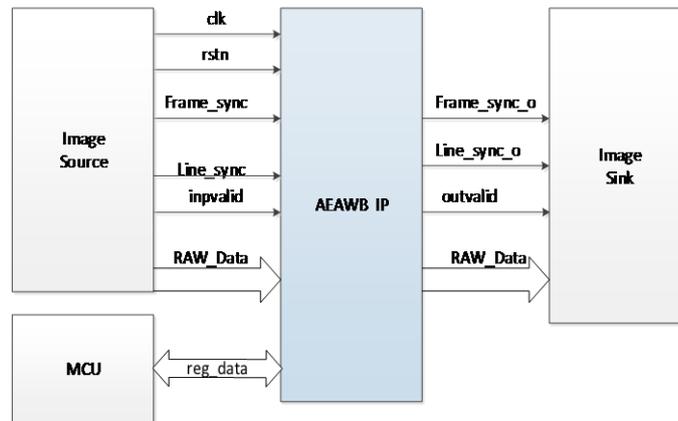
图 3-6 GAMMA 模块接口



3.4 AEAWB 模块

AEAwb，即自动曝光/自动白平衡。根据统计的图像信息对图像曝光度、白平衡调整处理。其模块主要接口如图 3-7 所示。

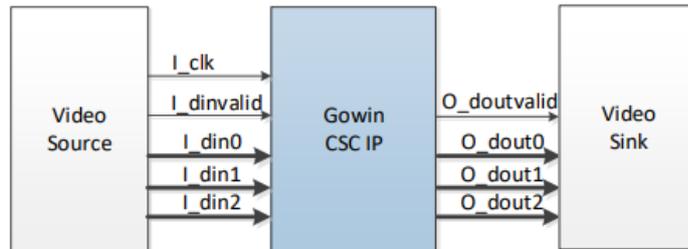
图 3-7 AEAwb 模块接口



3.5 CSC 模块

Gowin CSC (Color Space Converter) IP 用于实现不同的三轴坐标颜色空间 (RGB, YIQ、YCbCr) 转换, 常见的如 YCbCr 和 RGB 之间的转换。其模块接口如图 3-8 所示。

图 3-8 CSC 模块接口

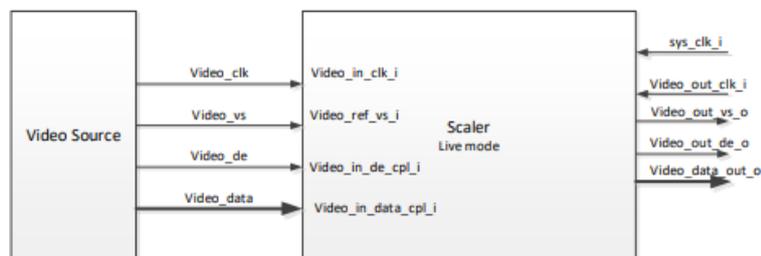


3.6 Scaler 模块

Gowin Scaler IP 用于把输入视频图像从一种分辨率 $X_{in} * Y_{in}$ 转换到另一种分辨率 $X_{out} * Y_{out}$ 输出。分辨率参数可在 IP 配置界面预先设置, 也可以动态实时配置, 另外 Scaler IP 所支持的插值算法包括最近邻插值, 双线性插值, 双立方插值。

其模块主要接口如图 3-9 所示。

图 3-9 Scaler 模块接口



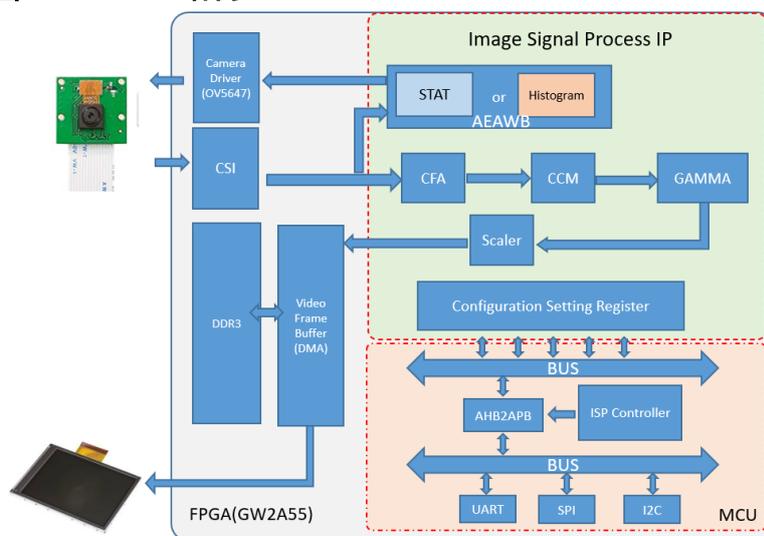
4 ISP 应用参考设计

4.1 Demo 设计采用的硬件平台

- ISP Demo 采用 DK_START_GW2A55-PG484 开发板进行实现。
- FPGA 部分主要实现图像传感器驱动控制，CSI 数据接收，ISP 图像信号处理，DDR3 图像存储，以及 LCD 驱动显示功能。
- MCU 部分通过 FPGA 的软核 Cortex-M1 实现，MCU 通过 UART 与 PC 的数据通信，实现指令控制和显示，之后数据指令通过 MCU 的 AHB 总线实现对 FPGA 的 ISP 模块的寄存器读写。
- 图像传感器型号为 OV5647，通过 FPGA 的 CSI 接口连接，实现控制和数据通信。
- LCD 型号为 AT070TN94，通过 FPGA 的 LCD 接口连接，实现图像显示。

4.2 Demo 结构设计说明

图 4-1 Demo 结构



FPGA 部分

主要通过驱动模块控制图像传感器，获取 1080p 的图像，ISP 模块处理 1080p 图像，Scaler 模块缩放 LCD 的图像大小，最后 LCD 显示图像，其中主要模块包括：

- Camera Driver 模块：实现对图像传感器驱动控制。
- CSI 模块：对传感器 CSI 过来的数据进行接收识别。
- CFA 模块：对来自传感器的 Bayer 图像数据转为 RGB 图像数据。
- CCM 模块：对 RGB 图像进行颜色校正。
- GAMMA 模块：对图像进行 Gamma 校正，使图像满足人眼对亮暗的敏感特性。
- AEAWB 模块：对 Bayer 图像进行数值统计以及直方图统计，根据统计数值进行白平衡处理，根据直方图信息完成曝光度的调节处理。
- Scaler 模块：对图像进行缩放。
- Video Frame Buffer 模块：对 DDR3 读写图像控制。
- DDR3：存储每帧图像数据。

MCU 部分

主要通过 PC 指令控制，FPGA 的数据通信，以及 AE 和 AWB 调试控制。

- PC 指令控制：通过 UART 与 PC 通信，每当 UART 终端发送指令，MCU 的 UART 中断响应，完成对指令进行接收，识别和发送操作。
- FPGA 数据通信：通过内部的 AHB 并行总线，对 ISP 模块的寄存器进行读写控制。
- AE, AWB 调试控制：通过读取 ISP 模块的统计信息和直方图信息，动态对 AE 和 AWB 的调节控制图像传感器，形成闭环控制。

4.3 MCU 指令控制

1. 初始化时，FPGA 中软核 Cortex-M1 将会通过 UART 打印相关初始化信息。



2. 指令控制，通过 UART 将指令格式发送给 FPGA 软核 Cortex-M1 进行相关控制。
- 指令格式为地址(12bit)+数据(32bit)+#(结束符号)。
 - 地址和数据为 16 进制数输入，间隔通过空格符号隔开。
 - 例子：如设置 AWB 寄存器值为 0x4ff，指令为 “ 0x104 0x4ff # ”
 - 寄存器读写操作如表 4-1 所示。

表 4-1 寄存器读写操作

	Name	Address	Data	W/R	Default	Note
ISP						
1	ISP_CMD	0x000	[3:0]	W	-	ISP command 4'b0001 --- show current mean value 4'b0010 --- show current histogram value 4'b0011 --- AWB processing 4'b0100 --- AE processing
2	ISP_REG_LOAD	0x004	[15:0]	W	-	Load reg's value from module
3	ISP_REG_GET	0x008		R		Get reg's value
Sensor						
1	CAM_AWB_EN	0x100	[0]	W	0x1	Sensor(Camera) AWB enable (0: Camera Auto enable ; 1: Manual enable)
2	CAM_AWB_R	0x104	[15:0]	W	0x0400	Sensor(Camera) AWB R gain value
3	CAM_AWB_G	0x108	[15:0]	W	0x0400	Sensor(Camera) AWB G gain value
4	CAM_AWB_B	0x10C	[15:0]	W	0x0400	Sensor(Camera) AWB B gain value
5	CAM_AGC_EN	0x110	[0]	W	0x1	Sensor(Camera) AGC manual enable (0: Camera Auto enable ; 1: Manual

	Name	Address	Data	W/R	Default	Note
						enable)
6	CAM_AGC_GAIN	0x114	[15:0]	W	0xFF	Sensor(Camera) AGC gain value
7	CAM_AEC_EN	0x118	[0]	W	0x1	Sensor(Camera) AEC manual enable (0: camera Auto enable ; 1: Manual enable)
8	CAM_EXPOSURE	0x11C	[19:0]	W	0x20	Sensor(Camera) AEC Exposure
CCM						
1	CCM_K11	0x200	[31:0]	W	0x0000142C	K11
2	CCM_K12	0x204	[31:0]	W	0xFFFFFC4C	K12
3	CCM_K13	0x208	[31:0]	W	0x000001A5	K13
4	CCM_K21	0x20C	[31:0]	W	0xFFFFF89	K21
5	CCM_K22	0x210	[31:0]	W	0x000015B5	K22
6	CCM_K23	0x214	[31:0]	W	0xFFFF760	K23
7	CCM_K31	0x218	[31:0]	W	0x00000108	K31
8	CCM_K32	0x21C	[31:0]	W	0x000000D9	K32
9	CCM_K33	0x220	[31:0]	W	0x000019B7	K33
10	CCM_O1	0x224	[31:0]	W	0xFF7AF699	O1
11	CCM_O2	0x228	[31:0]	W	0xFF7576A4	O2
12	CCM_O3	0x22C	[31:0]	W	0xFF7D23A3	O3
AEAWB						
	AEAWB_GET_AWB_GAIN_R	0x410	[31:0]	R		Get AWB_GAIN_1
2	AEAWB_GET_AWB_GAIN_G	0x414	[31:0]	R		Get AWB_GAIN_2
3	AEAWB_GET_AWB_GAIN_B	0x418	[31:0]	R		Get AWB_GAIN_3
4	AEAWB_MEAN_R	0x420	[31:0]	R		MEAN_R
5	AEAWB_MEAN_G	0x424	[31:0]	R		MEAN_G
6	AEAWB_MEAN_B	0x428	[31:0]	R		MEAN_B

	Name	Address	Data	W/R	Default	Note
	MEAN_B					
7	AEAWB_MEAN_Y	0x42C	[31:0]	R		MEAN_Y
8	AEAWB_SET_EXPOSURE	0x44C	[31:0]	W		Set the exposure to the MCU
9	AEAWB_HISTO_RDY_GET	0x460	[0]	R		Histogram is ready to read
10	AEAWB_HISTO_RGB_GET	0x500	[29:0]x256	R	-	Load histogram RGB value Each RGB with 10bit in FPGA's address from 0x500 to 0x5FF

3. 通过 UART 终端可以完成以下指令控制。

- 读图像 RGB 的平均值数据，指令：“0x000 0x01 #”

```
Set ISP's Reg: 0x0, 0x1
mean value of R: 40F583
mean value of G: 53E8CD
mean value of B: 139E
mean value of Y: 4C7AC2
```

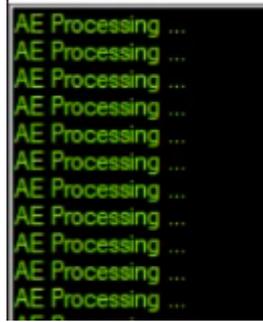
- 读图像 RGB 直方图数据，指令：“0x000 0x02 #”

```
Histogram Value: R G B
0: 420 21 0
1: 0 0 0
2: 0 0 0
3: 441 39 0
4: 0 0 0
5: 0 0 0
6: 0 0 0
7: 0 0 0
8: 0 0 0
9: 0 0 0
A: 0 0 0
B: 0 0 0
C: 0 0 0
D: 0 0 0
E: 0 0 0
F: 0 0 0
10: 0 0 0
11: 3 0 0
12: 3 400 3
13: 1F 0 A
14: 30 400 15
15: 45B 41F 3B
16: 96 406 67
17: 4D7 413 B3
18: 50C 3 CA
19: 4F8 4B F6
1A: 50C 83 11A
1B: 50C 4C3 13F
1C: 11C DE 12C
```

- AWB 自动白平衡处理，指令：“0x000 0x03 #”；开启后，若再次输入相同指令，将停止自动白平衡处理。

```
AWB Processing ...
```

- AE 自动曝光处理，指令：“0x00 0x04 #”；开启后，若再次输入相同指令，将停止自动曝光处理。



- 对图像传感器的相关寄存器写操作，“0x1XX 0xXX #”
- 对 ISP 的 CCM 模块的相关寄存器写操作，“0x2XX 0xXX #”
- 对曝光寄存器写操作，“0x44C 0xXX #”

