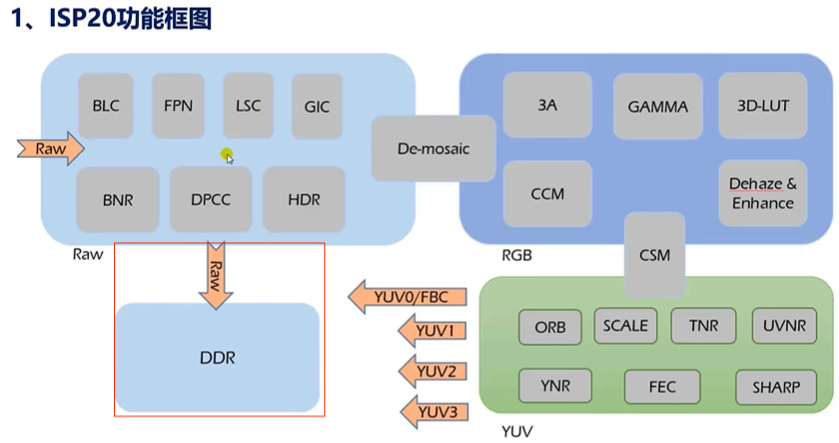
本节谈论ispserver是什么，有什么作用，以及怎么去测试。



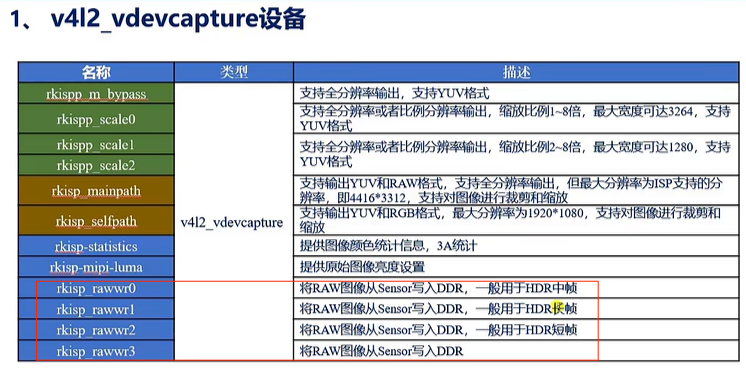
1. ISP20功能框图：



见参考文档：

"C:\Users\zhongqing\Desktop\笔记\08、RV1126参考资料\RV1126\_RV1109\Camera\Rockchip\_Tuning\_Guide\_ISP2x\_CN\_v1.6.2.pdf"

/\*



通过前面的上述PPT以及上面的图片可知：

ISP可以直接将Sensor输出的raw数据直接拷贝写入到DDR中。

\*/

/\*

ISP20经过De-mosaic模块后可以将raw图转化为RGB图。

再经过右上角和右下角的模块处理后，ISP20最终可以输出YUV/FBC格式的数据。

这里的YUV0 就是前面课程讲解的 rkispp\_m\_bypass节点对应输出的数据：



同理，YUV1、YUV2、YUV3就是后面 scale0、scale1、scale2输出的数据。

所以，raw数据的抓取，可以从mainpath节点（或者cif节点）抓取。

而rkispp\_m\_bypass、rkispp\_scale0、rkispp\_scale1、rkispp\_scale2节点对应的输出不再是raw格式。

\*/

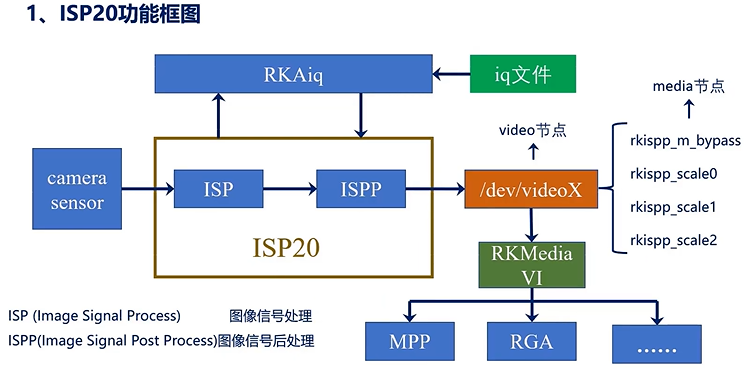
/\*

Rv1126的ISP20包括了硬件算法实现以及软件逻辑控制两个部分。

其中rkaiq就是软件逻辑控制部分。

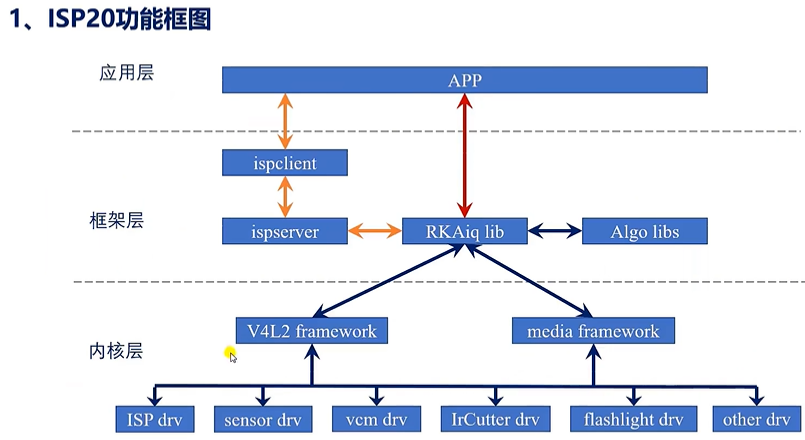
开发板就包含这个动态库： ./usr/lib/librkaiq.so 开发板中的这个就是软件控制部分，硬件控制部分就是rv1126的ISP硬件模块。也就是说rv1126的ISP20主要就包括硬件算法实现和软件逻辑控制。

我们首先来看rkaiq动态库部分：



摄像头的Sensor采集的数据经过ISP20采集之后，RKaiq动态库介入，rkaiq主要功能就是负责从isp驱动中获取图像统计信息。也就是获取图像统计数据。同时rkaiq也从iq文件在获取iq tuning参数，然后将图像统计数据以及iq tuning参数传递给算法库使用，例如传递给3A算法库或用户自定义算法库使用。这些算法库就进行计算操作。计算得到新的数据之后再把新的数据反馈给ISP20。ISP20得到新的数据之后再对图像进行调优处理。最终就可以从ISPP节点输出调优后的图像。

Rkaiq动态库的实现方式：



这个框图表达的就是：应用层操作RKAiq动态库有两种方式：使用橙色线路 或者 使用红色线路。两种方式二选一。

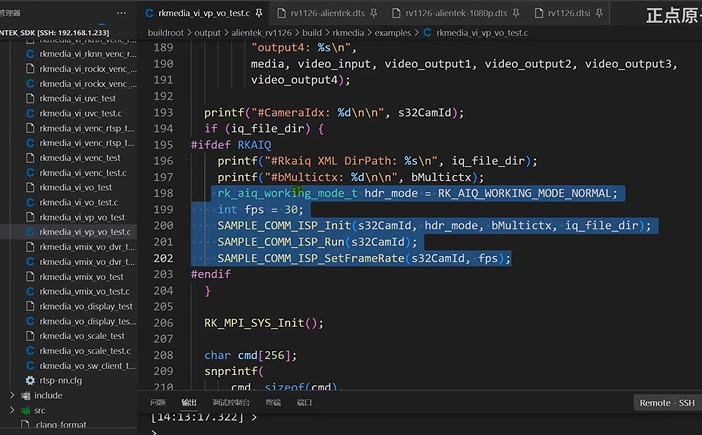
图中分为三层：

应用层：

框架层：主要就是rkaiq动态库的集成方式。它有两种集成方式：

橙色线路：称为ispserver的集成方式。就是在开发板的文件系统里去运行ispserver进程，然后isp客后端会与ispserver进行通信。上层的应用 就是 通过isp客后端到ispserver这条链路来连接rkaiq动态库的。Rkaiq动态库这边就会从驱动中获取图像统计数据并结合iq文件中的参数，将这些参数传递给Algo libs算法库（Algo libs就是算法库的意思）来进行计算。算法库计算好了之后再将新的参数反馈给下面的各个isp20涉及的硬件模块。Isp20的硬件模块就对图像进行调优，最后就可以输出效果较好的图像。（这种方式主要用于IPC应用 或 v4l2-ctl应用当中，这种方式的优点是：不用修改源码，只要开启ispserver进程之后，采集到的图像就具有isp调试的效果，也就是说得到的图像已经是经过3A处理之后的图片，所以是正常的。所以我们称ispserver是3A的后台守护进程）

红色线路：就是直接集成方式。这种方式就需要修改程序代码了。也就是调用RKaiq相关的API来实现的，在rkmedia下就有调用rkaiq的代码。例如：



它的这段程序就调用了rkaiq的API（SAMPLE\_COMM\_ISP\_Init）。

也就是说，想要得到经过isp处理后正常的图像，要么启动ispserver后台守护进程，要么直接集成修改程序代码。我们在测试IPC和使用v4l2捕获图像的时候，一般使用的就是开启ispserver，如果使用的是rkmeida的话，两条路选其一都可以走通。在前面的课程也有提到：



要么开启ispserver，要么指定iq文件修改程序代码，二者选其一。因为如果开启了ispserver进程就不能再走直接集成的线路了。

IMG_256

指定iq文件实际上就是走直接集成的线路。

1.出厂的文件系统是默认没有开启ispserver的，可以执行:

ps -aux | grep ispserver 查看ispserver是否开启

2.开启ispserver：

ispserver --no-sync-db & //表示不开启数据库服务dbserver

//或者

ispserver & &符号表示后台运行。

3.关闭ispserver服务：

kill -9 PID

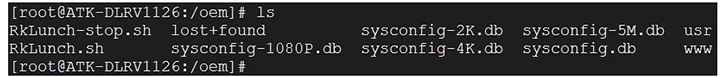
内核层：包含许多相关的驱动。例如：ISP驱动，摄像头Sensor驱动，VCM驱动（VCM就是音圈马达/音圈电机，主要用于调节镜头的位置，来改变焦距），IrCutter 驱动（就是红外滤波，用于过滤红外光）等等。

\*/

1. RKMedia 例程测试注意事项：

注意：如果开启了摄像头应用(mediaserver)，请先将其关闭后才可以使用 RKMedia抓取图片。

开启和关闭mediaserver的脚本在开发板的/oem目录下：



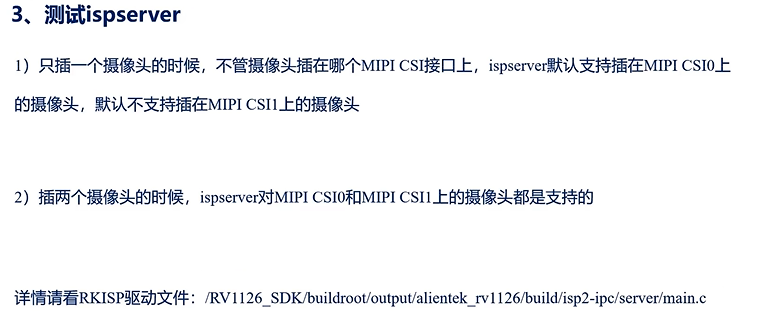
//开启摄像头应用：

/oem/RkLunch.sh &

//关闭摄像头应用：

/oem/RkLunch-stop.sh

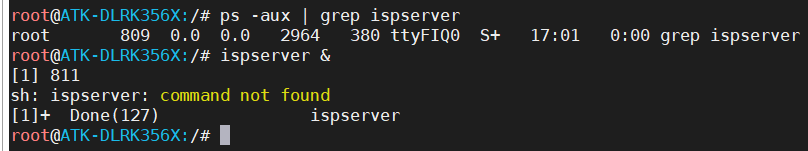
1. 测试ispserver:



1)只插一个摄像头的时候，不管摄像头插在哪个MIPI CSI接口上，ispserver默认支持插在MIPI CSIO上的摄像头，默认不支持插在MIPI CSI1上的摄像头

2)插两个摄像头的时候，ispserver对MIPI CSIO和MIPICSIl上的摄像头都是支持的详情请看RKISP驱动文件：

/RV1126\_SDK/buildroot/output/alientek\_rv1126/build/isp2-ipc/server/main.c



我们进入/etc/iqfiles 目录下面都是iq文件

