

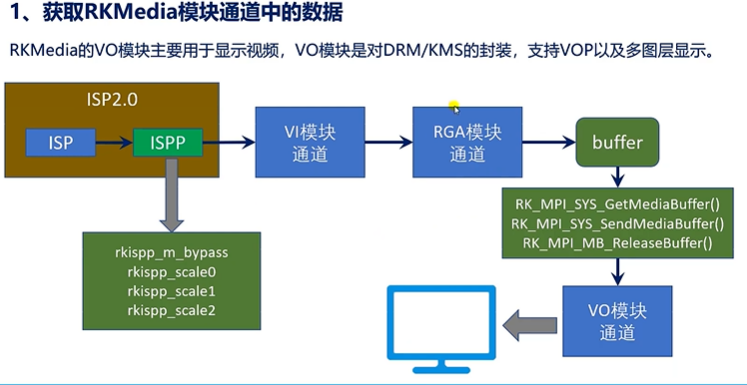
RKMedia是以模块化的方式来进行设计的。模块有输入接口和输出接口。可以通过绑定通道的方式控制数据流从某个模块的通道流出再流入另外一个模块的通道中。

例如：通过通道绑定的方式，我们可以将Vi模块的通道与RGA模块的通道进行绑定，然后RGA模块的通道再和VO模块的通道进行绑定。那么我们调用VO模块对应的接口就可以间接地从ISPP的四个节点获取码流。

VI模块获取到的码流通过绑定的通道发送给RGA模块进行处理，RGA模块主要就是对图像进行旋转、色彩空间转换、缩放和裁剪操作的。RGA处理好后再通过绑定的通道将码流发送给VO模块。VO模块就利用DRM显示框架把图片显示出来。

以上就是通道绑定的方式，将一个模块通道中的数据发送到另一个模块的通道中，也就是说，如果某一个模块的通道想要获取另外一个模块中通道的数据的话，就可以通过通道绑定的方式来实现。

或者也可以不通过通道绑定的方式获得数据流：

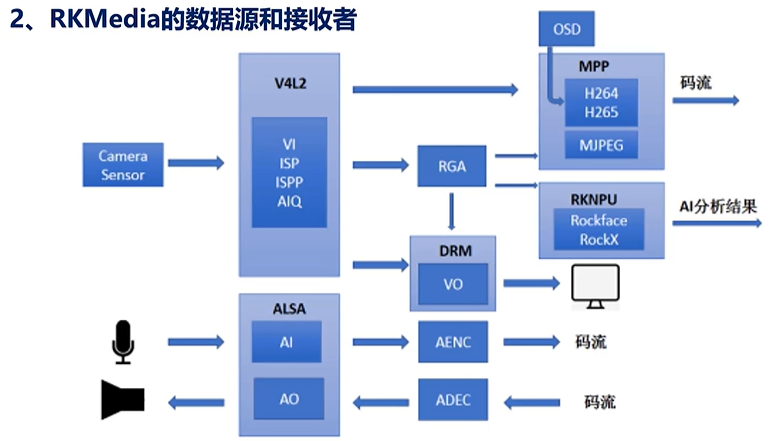


将RGA模块输出的数据先放在一个framebuffer中，其它模块再从这个buffer中获取图像数据。将framebuffer中的图像发送给另外一个模块的话，可以调用RKMedia的这三个RK\_MPI系统控制函数:

首先调用RK\_MPI\_SYS\_GetMediaBuffer这个API来获取对应的模块存放在这个buffer中的数据。

再调用RK\_MPI\_SYS\_SendMediaBuffer把获取到的数据发送给另外一个模块。

最后还得调用RK\_MPI\_MB\_ReleaseBuffer来释放之前占用的资源。

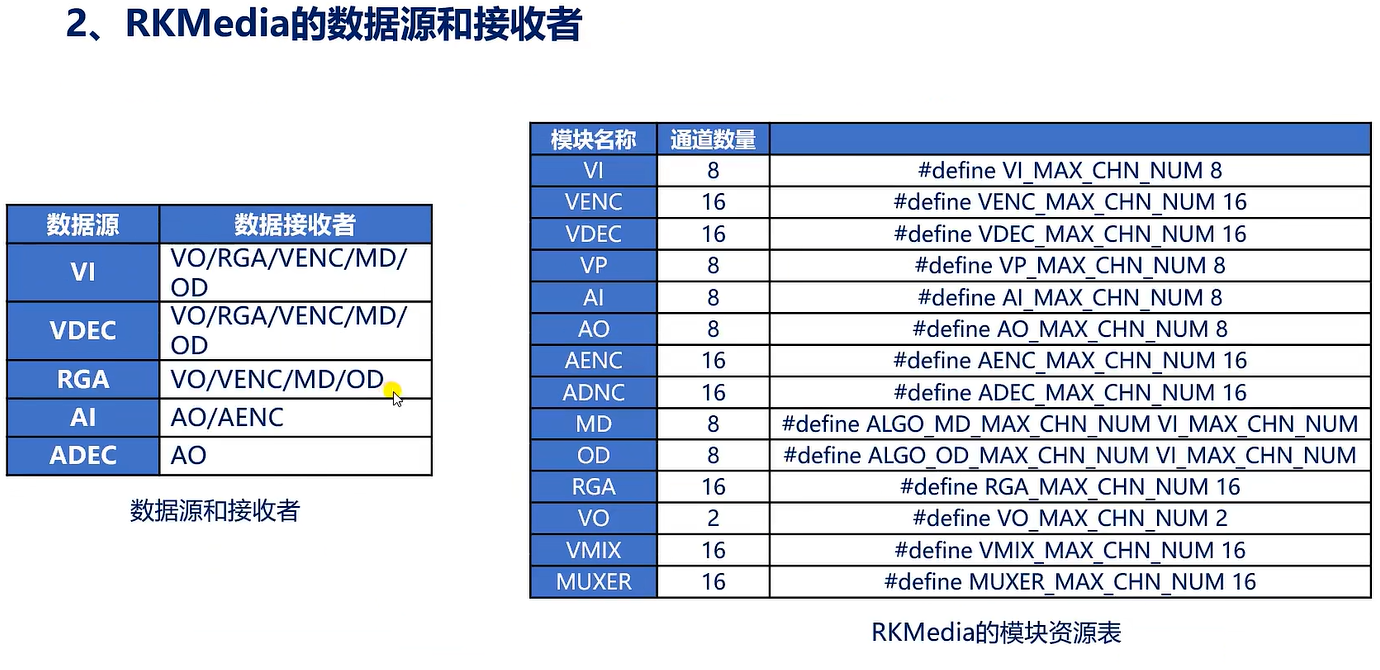


根据手册中的这个图里箭头的指向，我们就可以知道。数据可以从哪个模块流出然后流入到哪个模块在。

在通道绑定中，输出码流数据的模块称之为数据源。

接收码流的模块被称之为接收者。

如下的表格就列出了：数据源和接收者可以是哪些模块。

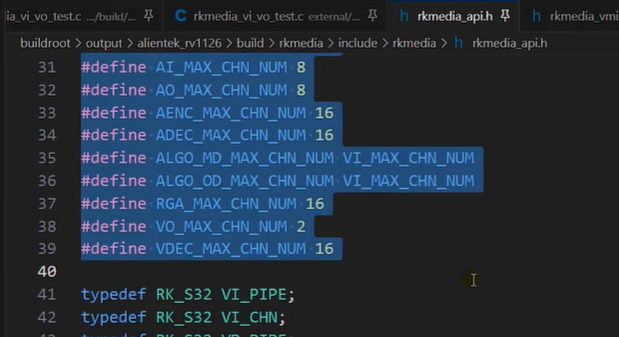


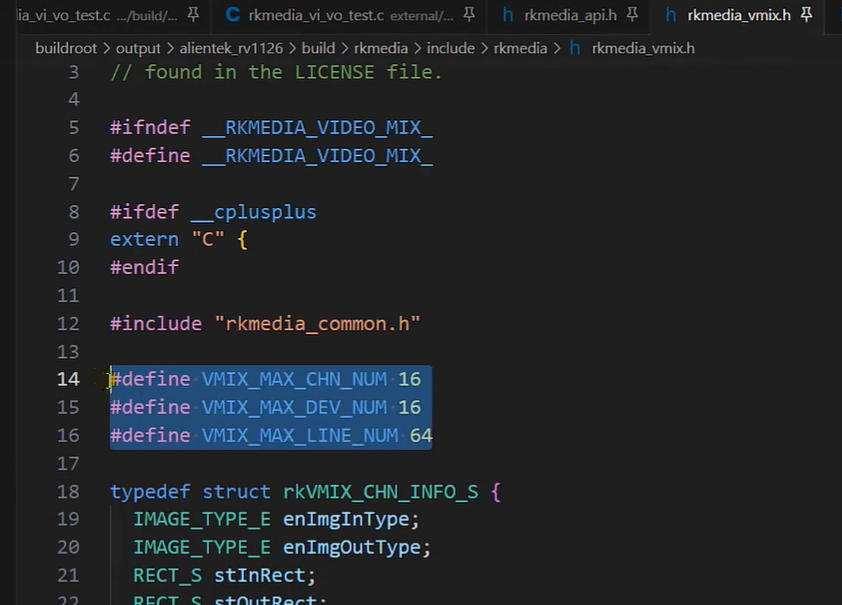
在后面编写代码 进行绑定通道的时候，就需要根据该手册来编写。

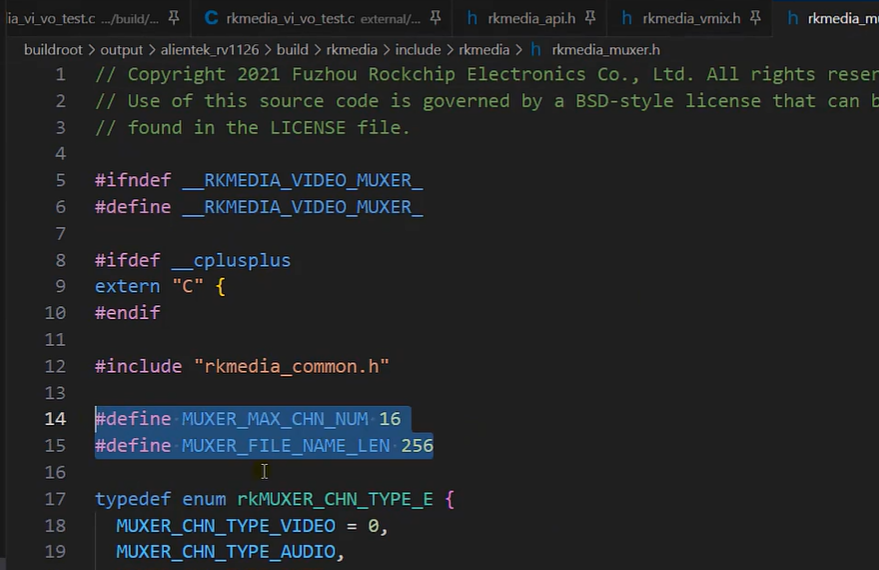
(哪个模块的通道可以绑定哪个哪个模块是有要求的。)

模块的通道数量如右图，也可以根据需要修改宏定义。

/\*





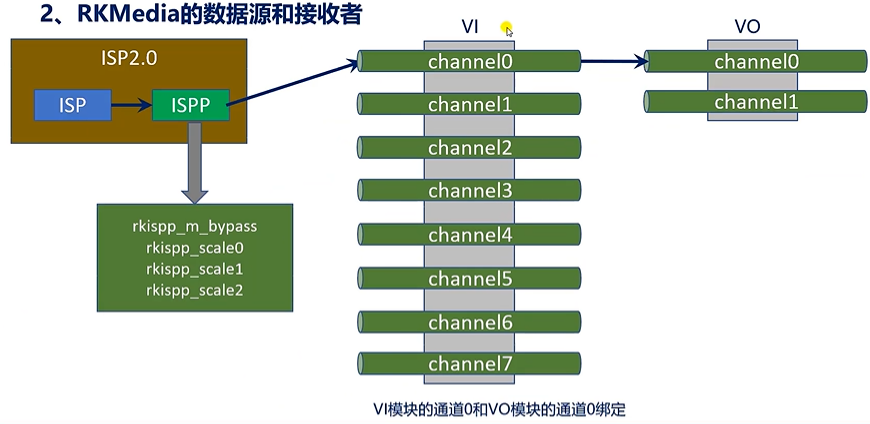


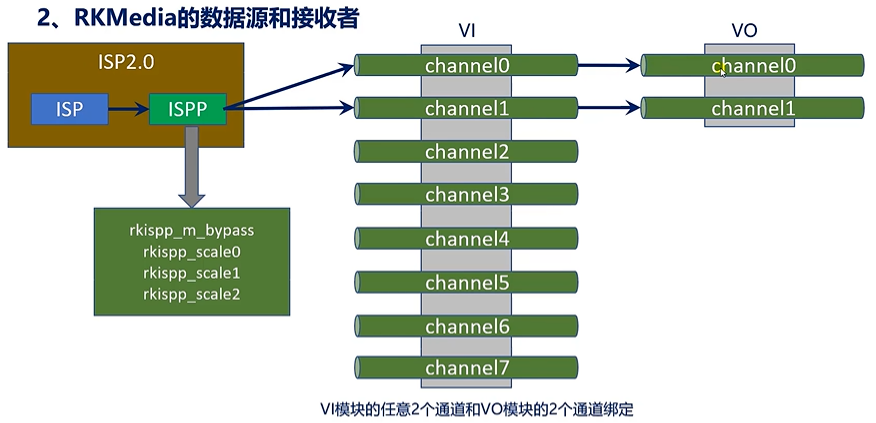
宏定义可以在如上几个文件中找到。

\*/

/\*

这里通道可以理解为：



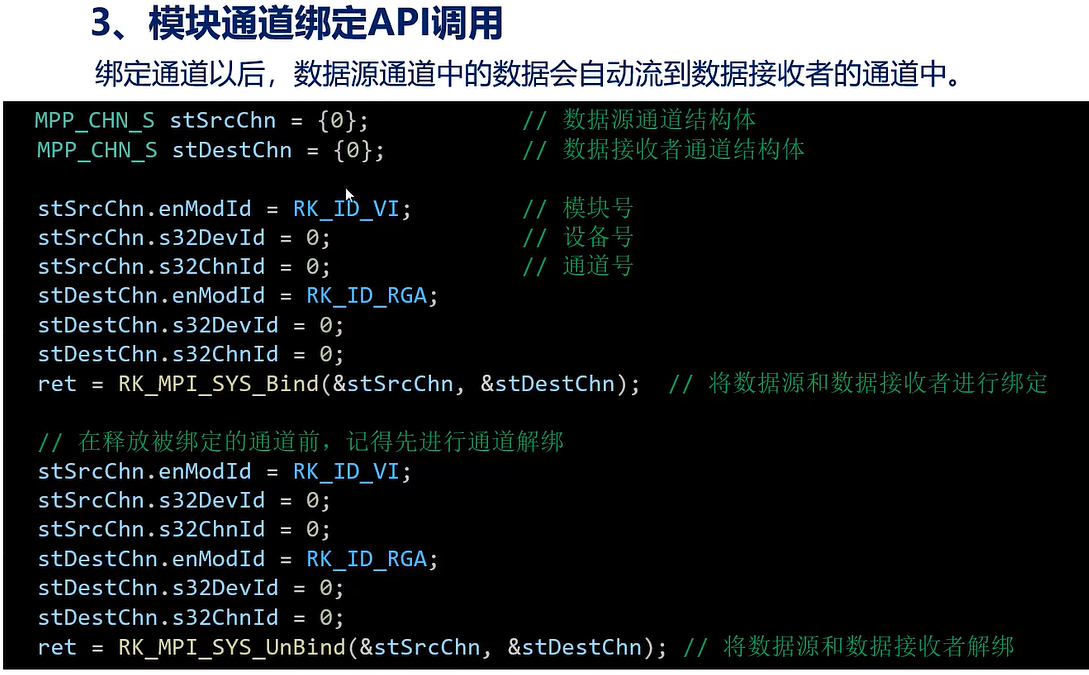


一个通道不能同时绑定两个或多个通道。只能一对一绑定。不能一对多绑定。也不能多对一绑定。

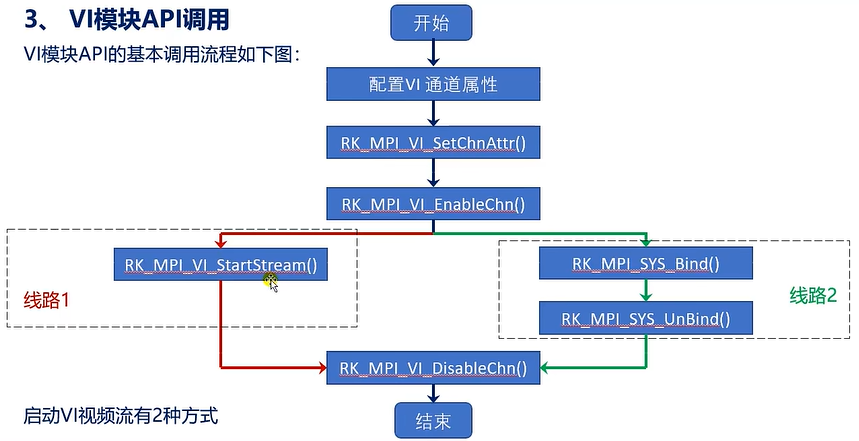
\*/

/\*

通道绑定的API:



声明并初始化数据源通道的结构体 和 数据接收者通道的结构体。



我们由此图可知：VI模块的通道在建立通道和使能通道之后，VI的数据还不能流出来，还得启动VI通道的数据流（RK\_MPI\_StartStream）才可以流出来。

启动VI的数据流就有两种方式。调用API实现和通道绑定的方式实现。只有AI模块和VI模块才有StartStream这种api。

释放资源或者设置通道属性的时候是需要先解绑通道的。

\*/