Platform: ROCKCHIP

OS: Android 6.0

Kernel: 3.10.92

上一篇小结了UVC协议, 大概理解之后再看driver相对会轻松些.

uvc\_init -> uvc\_driver.c

    usb\_register ->    //注册到usb core

        uvc\_probe ->    //摄像头插入,枚举成功后被调用

            uvc\_parse\_control -> //解析除了四大类描述符之外的所有描述符,四大类描述符在usb枚举的时候已经解析过了.

                uvc\_parse\_vendor\_control    //解析vendor-specific控制描述符,这里貌似只有Logitech厂商的.

                uvc\_parse\_standard\_control ->    //包括VS, 所有的entities,比如IT,OT,PU,SU等, Unit和Terminal会添加到uvc\_device的entities链表.

                    uvc\_parse\_streaming    -> //先解析VC HEADER,即Class-Specific VC Interface Descriptor

                    //, 里面会包含VC的总长度,当前版本,以及有几个VS, 然后循环调用uvc\_parse\_streaming解析VS.

                    //包括VS HEADER, frame, format等, 主要填充uvc\_streaming结构体.最终被添加到uvc\_device的streams链表.

                        uvc\_parse\_format    //解析format描述符.

            v4l2\_device\_register    //注册成为标准的v4l2 device,对于Camera HAL来说,uvc和一般camera是一样的行为.

            uvc\_ctrl\_init\_device ->    //根据前面得到的不同的entities相关信息都放到struct uvc\_control中管理.

                uvc\_ctrl\_init\_ctrl    //从硬件得到的CTRL和代码中原先的配置和mappming做匹配,然后添加到uvc\_device中.

            uvc\_scan\_device     -> //针对不是OT的entities创建struct uvc\_video\_chain并绑定起来.

                uvc\_scan\_chain    //entity和video chain绑定.

                list\_add\_tail    //添加到uvc device中管理.

            uvc\_register\_chains ->    根据前面得到的chain后进行注册

                uvc\_register\_terms ->    //只有type是UVC\_TT\_STREAMING的才能注册,也就是说一个stream对应注册一个video.

                    uvc\_register\_video ->

                        uvc\_video\_init

                            uvc\_queue\_init    //video buffer queue初始化

                            usb\_set\_interface

                            uvc\_get\_video\_ctrl    //把设备的配置作为stream control的默认参数.

                        video\_device\_alloc

                        video\_register\_device    //注册的v4l2 device对应的ops是uvc\_fops

            uvc\_status\_init    //USB input/urb相关内容初始化

                uvc\_input\_init    -> //camera是作为一个input设备存在的.

                    input\_allocate\_device

                    input\_register\_device

                usb\_alloc\_urb

                usb\_rcvintpipe        //创建接收中断endpoint.

                usb\_fill\_int\_urb    //urb填充

---------------------