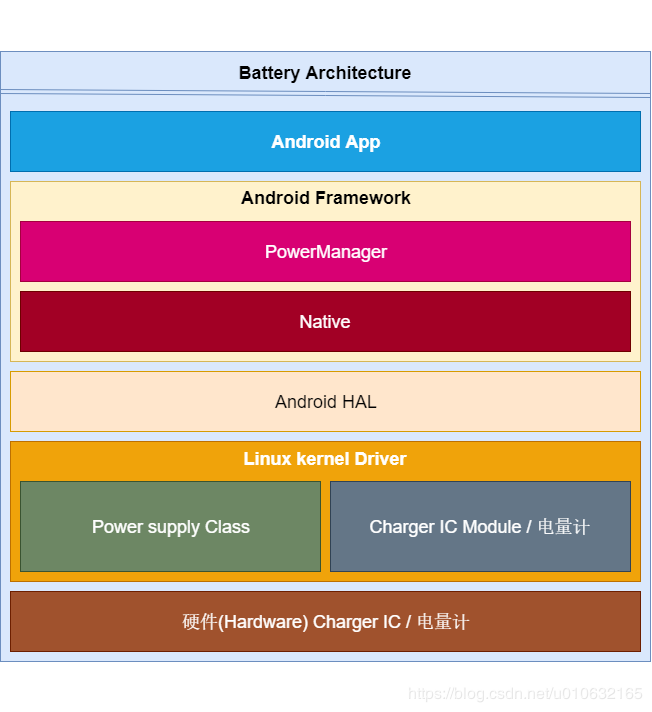
1. 整体架构



一、 内核层

1. 硬件驱动:

充电IC驱动: 主要包含充电IC初始化、提供充电控制接口及其他充电通用驱动要 求的接口

电量计驱动: 主要包含电量计初始化、提供获取电池状态接口及其他电池通用驱动 要求的接口

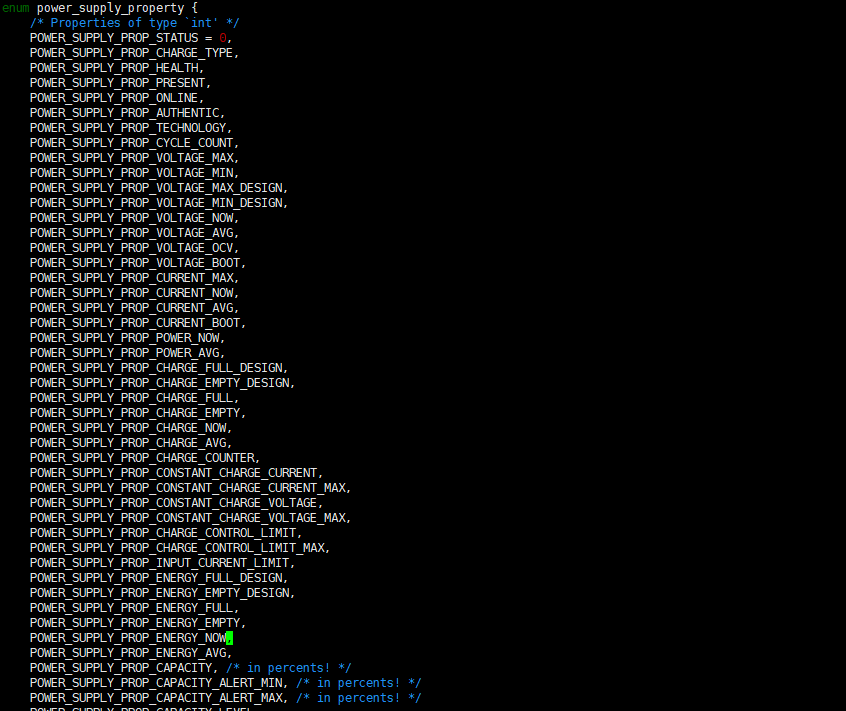
2. 充电通用驱动:

1. 向power\_supply子系统注册psy设备, 比如AC, USB, Wireless

2. 充电控制及监控, 主要包含充电电压，充电电流等

3. 充电安全控制，主要包含输入过压、输入过流、温控等

4. otg供电，此为非必要选项

5. 充电状态上报， 具体上报参考include/linux/power\_supply.h 

电池通用驱动:

1. 向power\_supply子系统注册psy设备, 比如Battery
2. 电池安全监控: 主要包含电池过压、过量、过温等
3. 电池状态信息上报

Power\_supply子系统:

将某个PSY设备支持的属性及其value，以sysfs的形式，提供给用户空间；当属性值改变时，以uevent的形式，广播给用户空间程序。

1. Power supply core，用于抽象核心数据结构、实现公共逻辑。位于drivers/power/supply/power\_supply\_core.c中。
2. Power supply sysfs，实现sysfs以及uevent功能。位于drivers/power/ supply /power\_supply\_sysfs.c中。
3. Power supply leds，基于linux led class，提供PSY设备状态指示的通用实现。位于drivers/power/power\_suppply\_leds.c中。

常用函数：

power\_supply\_register()

power\_supply\_get\_by\_name()

power\_supply\_get\_property()

power\_supply\_set\_property()

power\_supply\_changed()

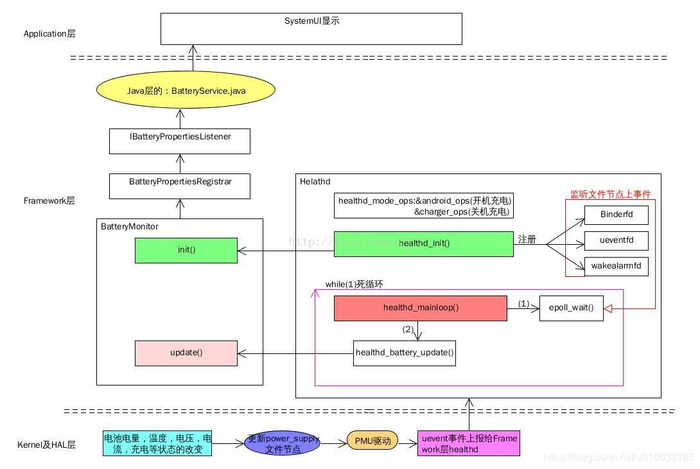
Android Hal:

这个模块主要是通过监控uevent来实时获取内核上报的电池充放电信息，并通过call\_back回调函数，将获取的充放电信息传传给framework层

主要代码：system/core/healthd/

hardware/interfaces/health/

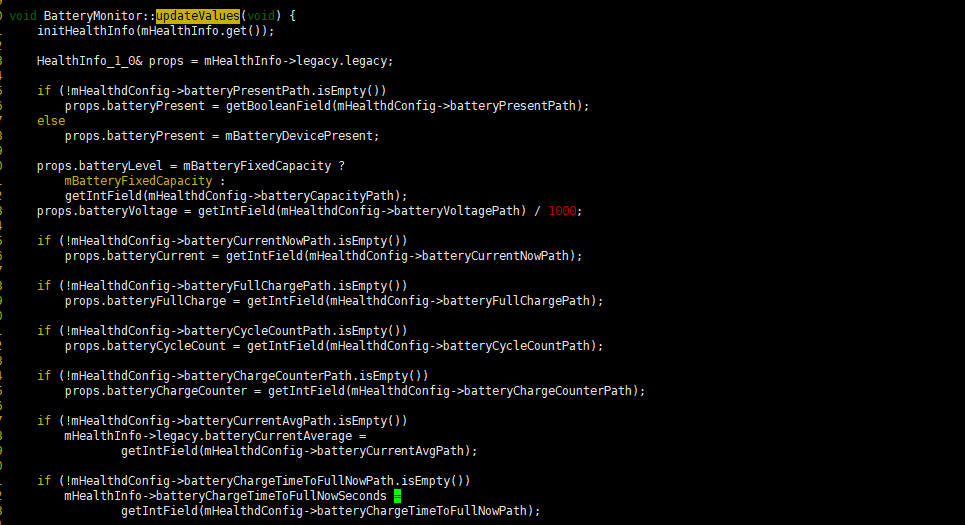
详细参考：https://note.youdao.com/s/EYwDlIWI



Hal 调用栈:

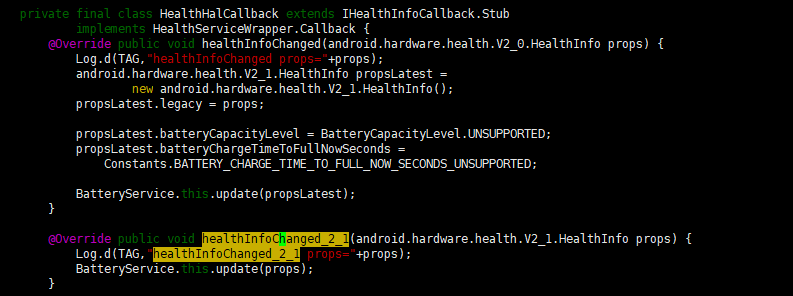


获取电池充放电信息主要函数方法：

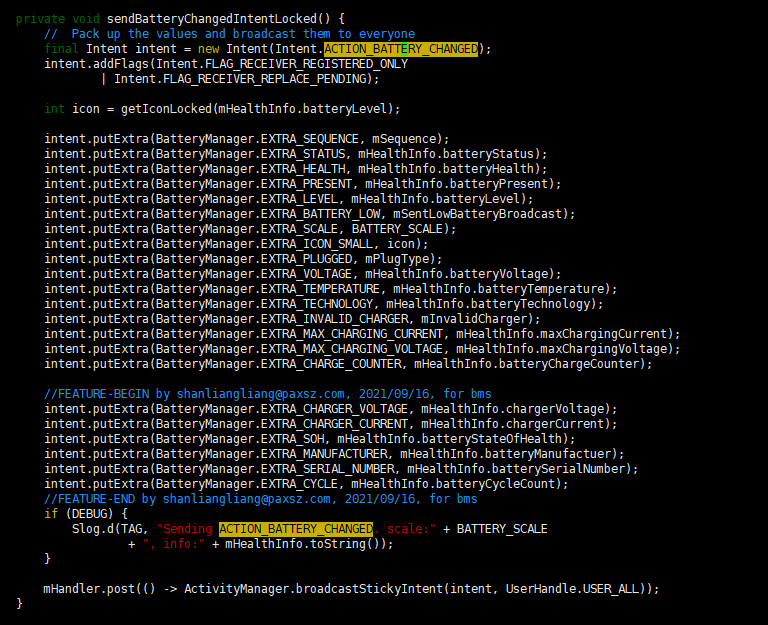


Android Framework层:

1. 用AIDL跨进程方式向**healthd service**注册回调函数



1. 通过Intent.ACTION\_BATTERY\_CHANGED广播向framework其他模块和应用更新充放电信息 frameworks/frameworks/base/services/java/com/android/server/BatteryService.java



Android APP: Setting, system UI,以及第三方apk