1. QCRIL在初始化过程中为什么还要将旧的QCRIL机制进行初始化？

QCRIL在初始化过程中将旧的QCRIL机制进行初始化主要目的应该是一种兼容,另外保留旧的机制也能在扩展某些不易于模块化的流程时提供一种解决方案.

1. QCRIL各个Module是如何实例化的？

QCRIL各个Module初始化主要在qcril\_init()函数中进行,该函数中还会对Event table进行初始化.

1. RIL层是如何启动QCRIL的?

rild.c的main()函数中通过dlHandle = dlopen(rilLibPath, RTLD\_NOW)来动态加载reference-ril.so或者qcril.so.然后通过

rilInit = (const RIL\_RadioFunctions \*(\*)

(const struct RIL\_Env \*, int, char \*\*))dlsym(dlHandle, “RIL\_Init”);

来获取so库中的RIL\_Init函数,接着通过以下调用

funcs = rilInit(&s\_rilEnv, argc, rilArgv)

来获取so库中回调函数列表,并完成qcril中mainLoop的建立,最后通过调用

RIL\_register(funcs);

来完成回调函数列表的注册.

1. funcs = rilInit(&s\_rilEnv, argc, rilArgv);这个函数中初始化了一个重要的结构体是哪个，返回的对象即funcs表示什么?

初始化的重要结构体是RIL\_Env,该结构体实际上定义了qcril向上response时的回调函数列表.funcs是qcril返回给rild的回调函数入口,作为rild下发request的入口.

1. 【附加题】画出setup data call 的时序图，时序图中要包含RILJ、rild、QCRIL具体对象间的消息流转和主要处理函数。

