1、程序架构：

1.1: Lib(总库）

**1.1.1:Sdk**（对应“Prolin应用编程接口编程指南(V2.3.3)”）

**Sdk**（使用库文件夹，包括so文件与相应的头文件）

**sdk\_src**（库原码，包含c文件、h头文件、Makefile\Android.mk,

编译输出在上面sdk文件中）

**1.1.2**:Gui（对应“Prolin XUI接口说明(V2.1.6)”另加 KS API\_GUI接口（主要自用））

gui（使用库文件夹，包括so文件与相应的头文件）

gui\_src（库原码，包含c文件、h头文件、Makefile\Android.mk,

编译输出在上面gui文件中）

**1.1.3:**Tls（最新版本的mbedtls 另加 引用接口与SM4功能）

tls（使用库文件夹，包括so文件与相应的头文件）

tls\_src（库原码，包含c文件、h头文件、Makefile\Android.mk,

编译输出在上面tls文件中）

1.2: Main(守护进程，安装主控应用，管理应用参数与应用间切换）

安装主控时验签用到了Tls库，创建共享内存进入主控应用与切换默热应用。

1.3: Master（主控应用，主要用于终端参数管理，客户应用更新与安装，

附加工厂测试功能。）

安装客户应用时验签用到了Tls库，界面显示用到了Gui，基本的应用列表查看与应用间切换其它功能都用到了sdk库。

1.4: App（客户应用，以sdk为基本，以app\_main 为入口，gui与tls 为选配库，

自由开发）

1.5：签名打包（用于主控与应用的打包与签名）

1. 目录结构：

+runtime（基础目录）

start.sh (启动脚本）

+config (适配不同的硬件平台)

+daemon (守护进程）

+tms (ota)

+master (主控应用）

+sdk (文件夹）

libsdk.so (sdk库）

libgui.so (gui库）

libtls.so (tls库）

+res (ks.res\\*)

+data ()

+app

+app\_id ()

+sys (bin\log)

+data (app create file “filename.ext”)

+app\_id

3、开机流程：

start.sh->守护进程->主控应用->客户应用