1、将手机root；如果手机不root，关闭不了SELinux，IDA如果附加进程调试，则看不到运行的进程。

2、关闭SELinux

adb shell getenforce

adb shell setenforce 0 //设置成permissive 模式

adb shell setenforce 1 //设置成enforce 模式

注意此方法重启后失效

3、将IDA安装目录下的dbgsrv/android\_server拷贝到手机中，修改文件权限

adb push android\_server /data/local/tmp

adb shell chmod 777 /data/local/tmp/android\_server

4、安装添加了android:debuggable="true"属性的apk到手机

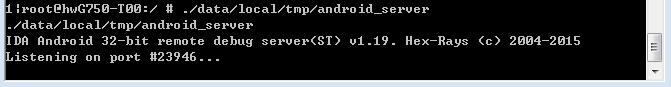
在AndroidManifest.xml的application中添加android:debuggable="true"。

adb install my-boy-free-gba-emulator-1-8-0\_killer.apk

注意：使用的是Android\_killer v1.3.1.0对原apk进行反编译，注意bin/apktool/apktool/ShakaApktool.jar要是最新的，不然反编译会失败。

5、启动android\_server

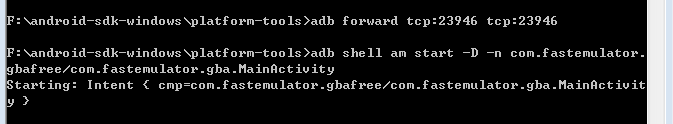
./data/local/tmp/android\_server



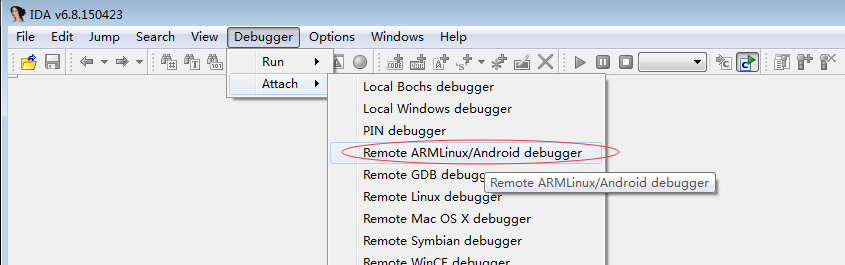
6、在启动一个命令行，端口转发，启动调试

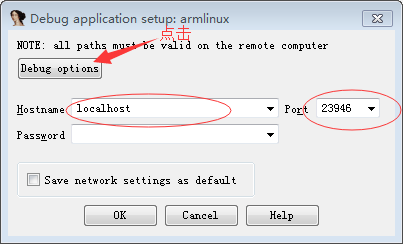
adb forward tcp:23946 tcp:23946

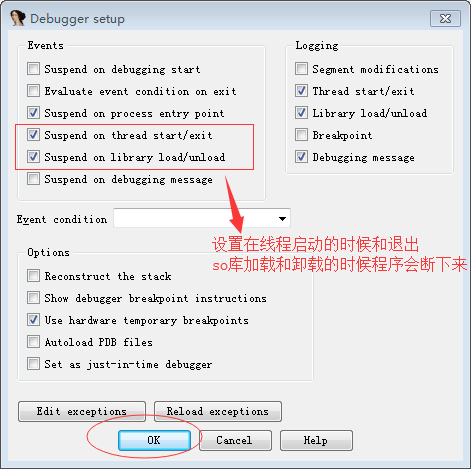
adb shell am start -D -n com.fastemulator.gbafree/ com.fastemulator.gba.MainActivity

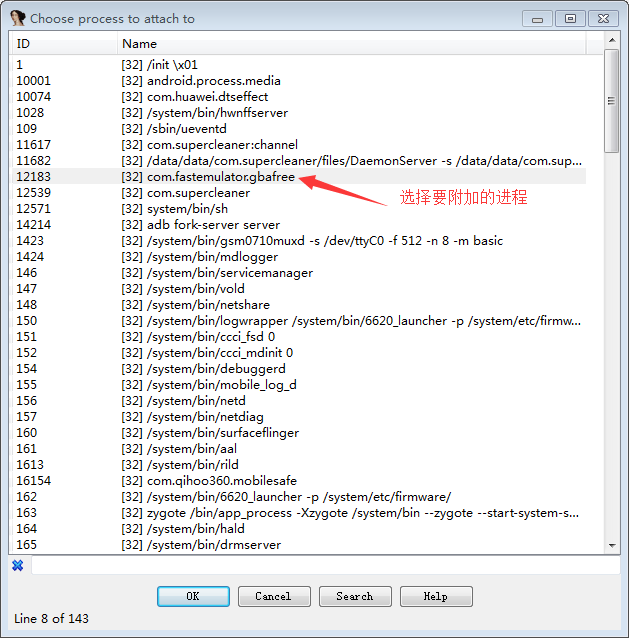


7、打开IDA，附加上对应的进程之后，设置IDA中的load so时机，即在debug options中设置



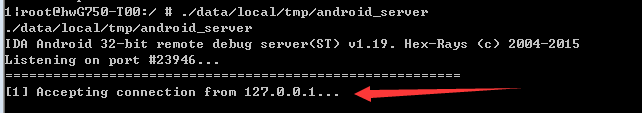


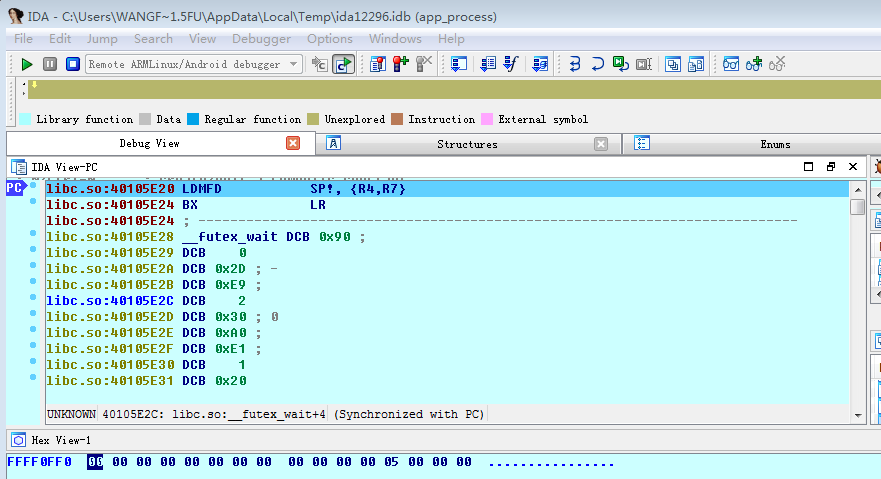




这里选择附加的进程可以使用Ctrl+F查找进程

附加进程后的效果：





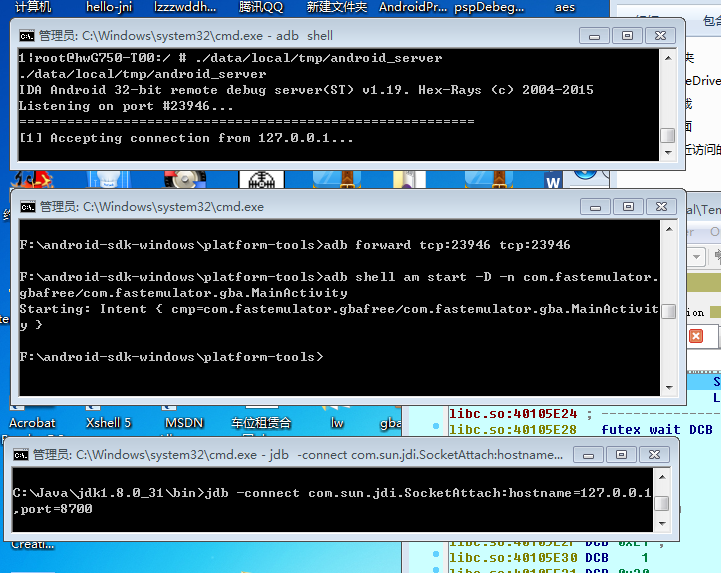
手机上面现实的是Waiting for Debugger

8、在启动一个命令行，

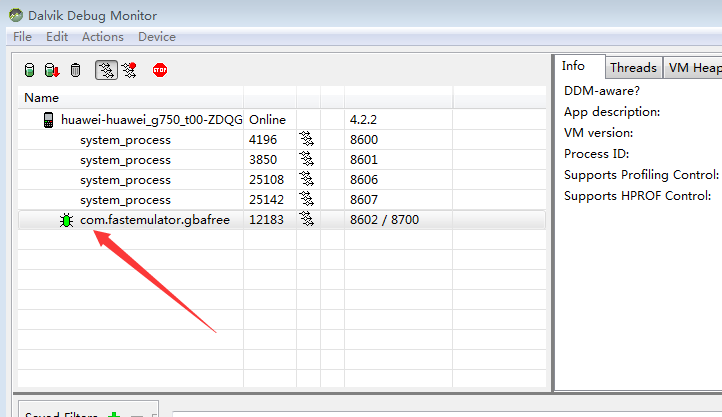
jdb -connect com.sun.jdi.SocketAttach:hostname=localhost, port=8700



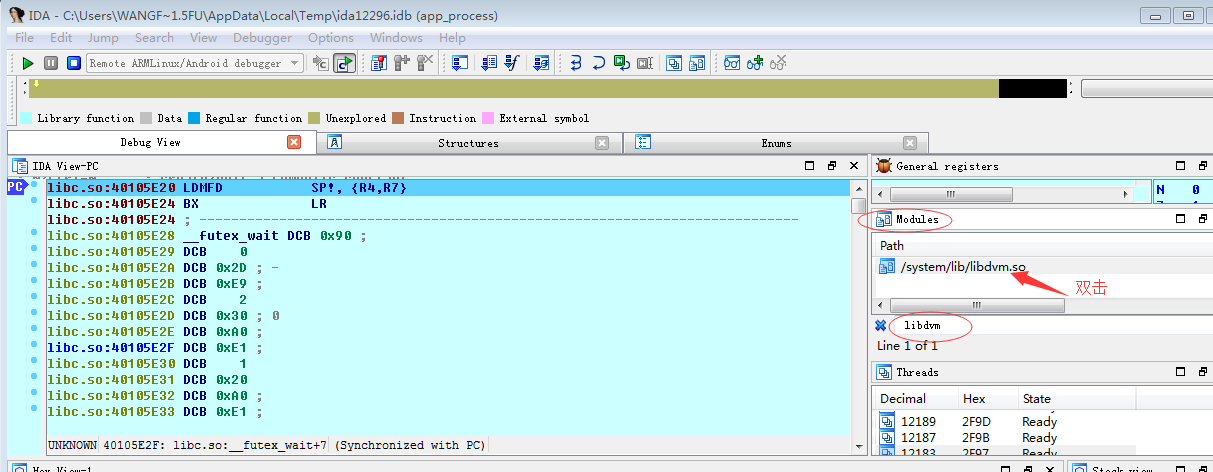
注意此时一共有三个控制台

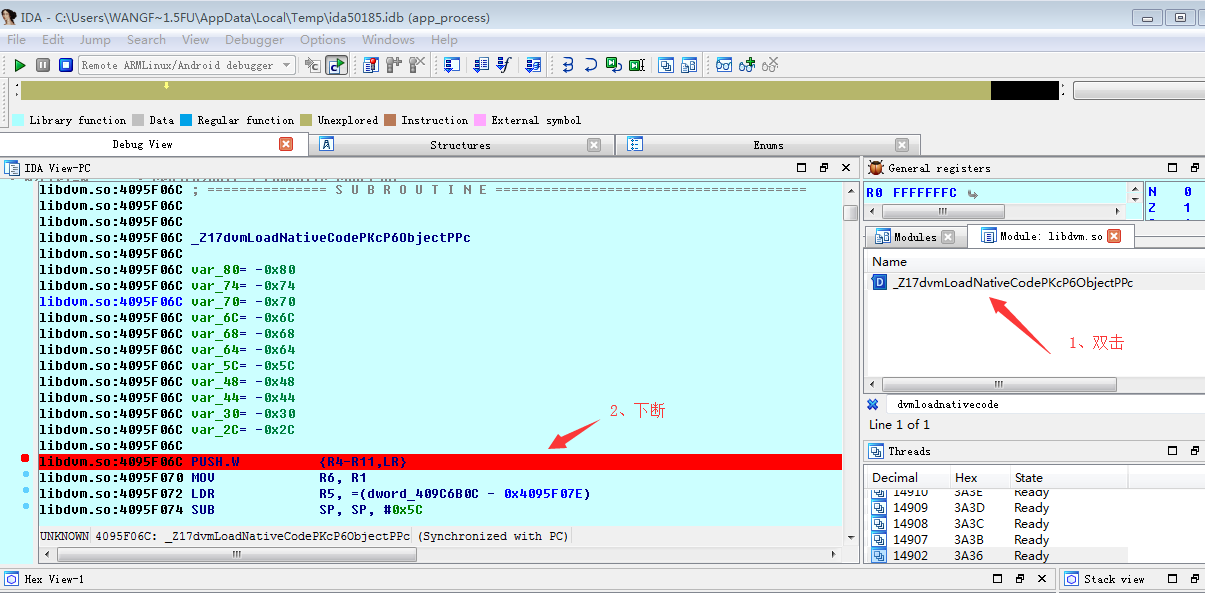


9、此时打开ddms，可以看到应用的状态是调试

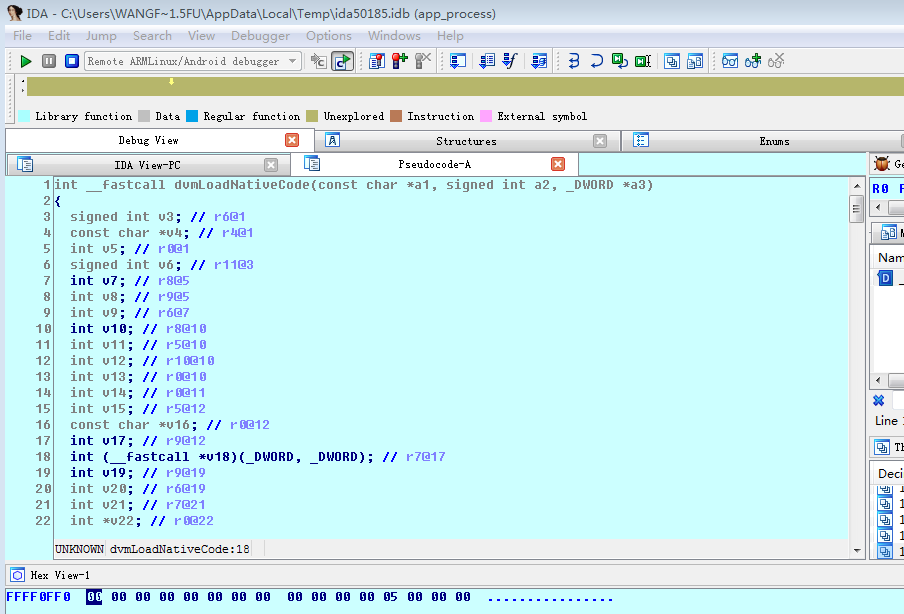


10、查看libdvm.so里面的符号dvmloadnativecode，在dvmloadnativecode处和此函数的调用JNI\_Onload的地方设置断点

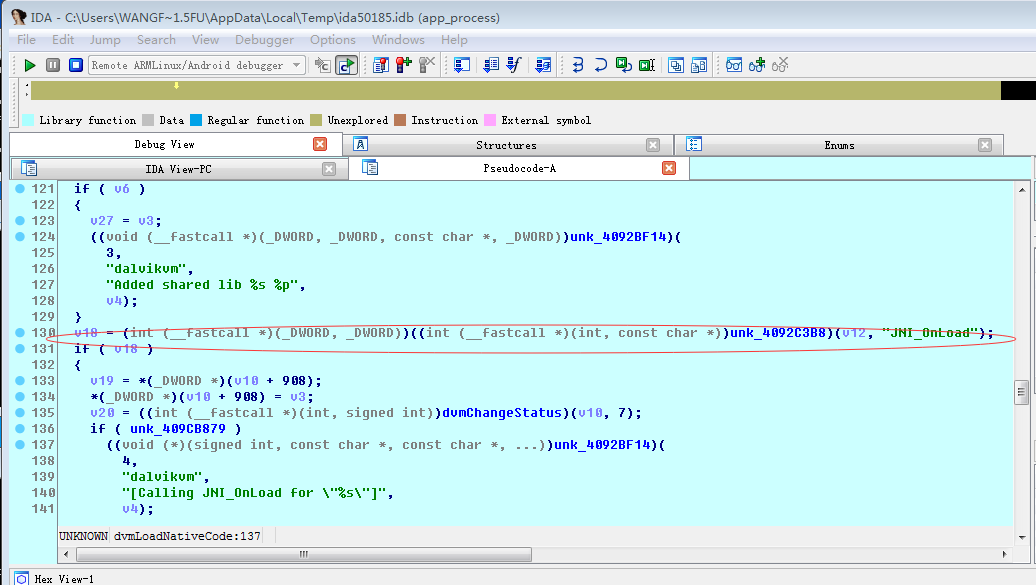




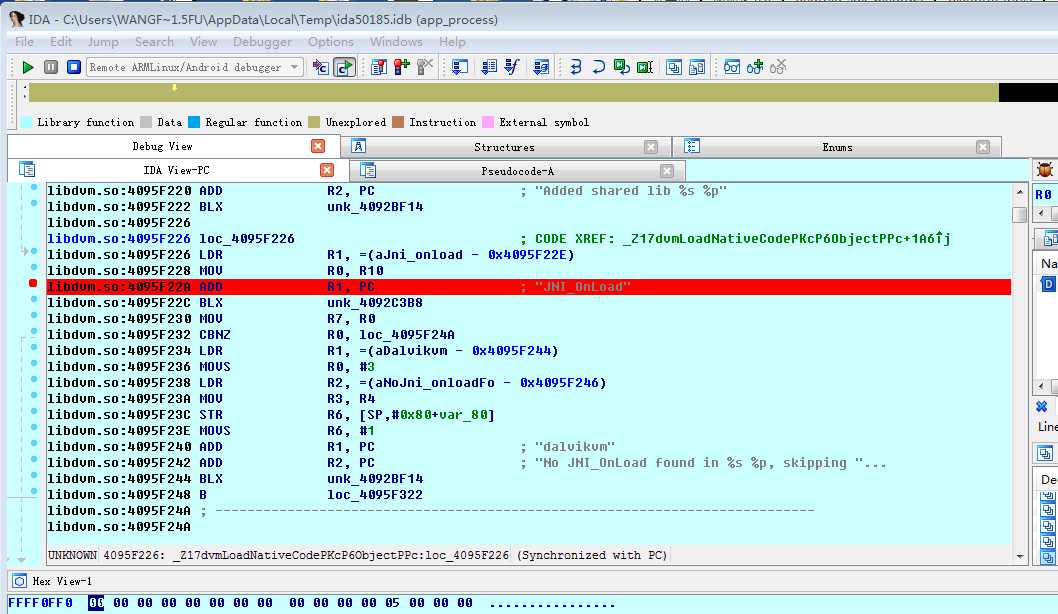
F5跳转到伪代码



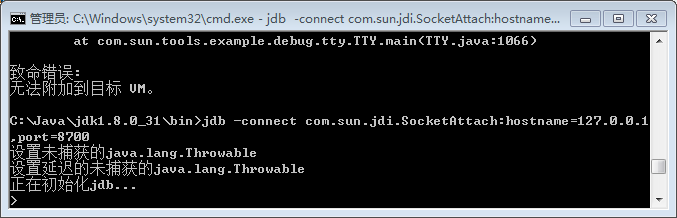
找到JNI\_Onload调用的地方



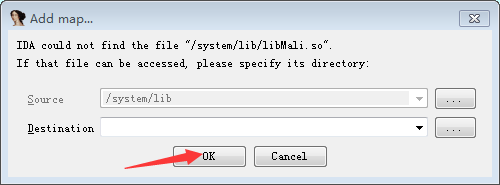
Tab键跳转到汇编代码，下断，如果不在此处下断点，调试时，会找不到libgba.so的函数符号

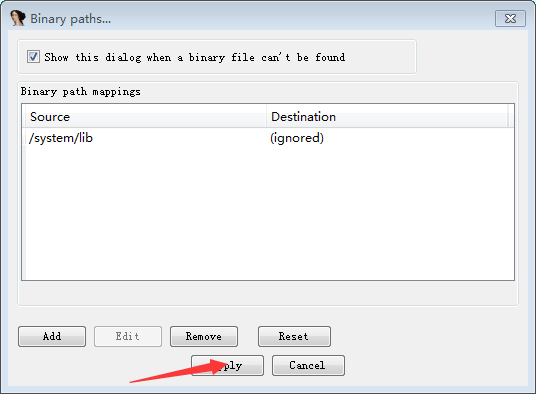


IDA快捷键F9运行

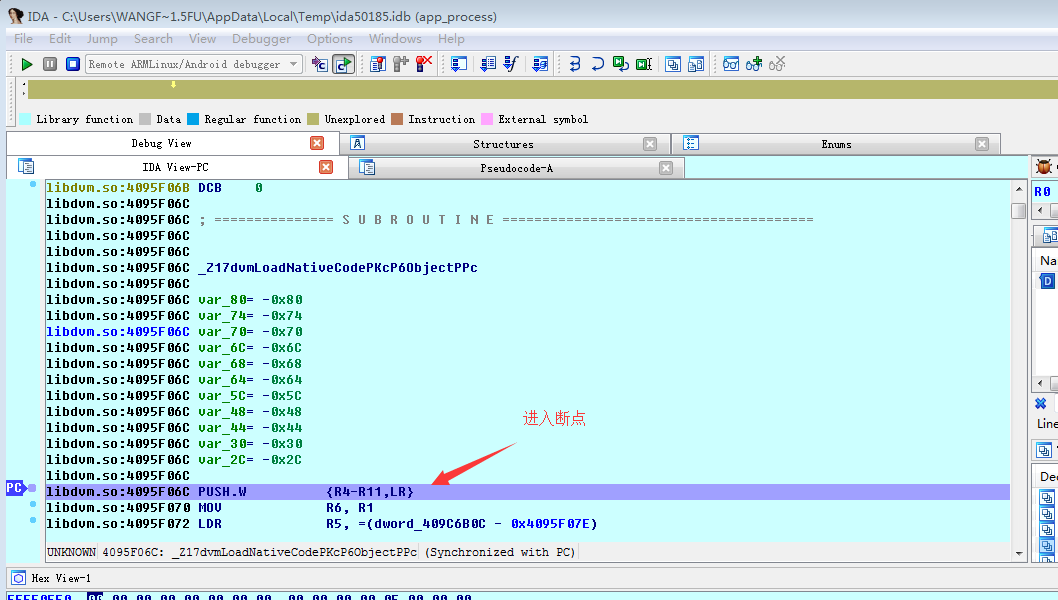


此时IDA会弹出一些弹窗，直接点击ok，apply，此处这个提示不知道为什么，实际手机目录下面有so。

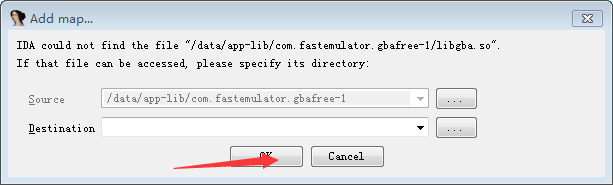


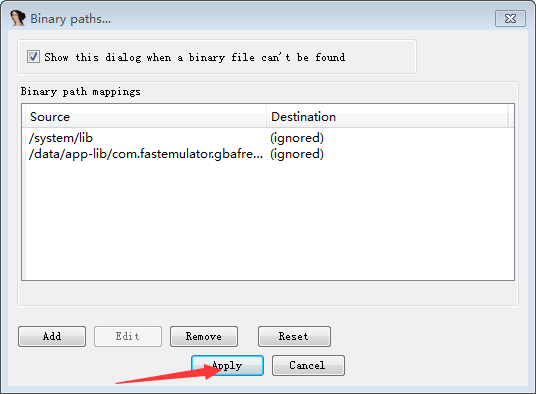


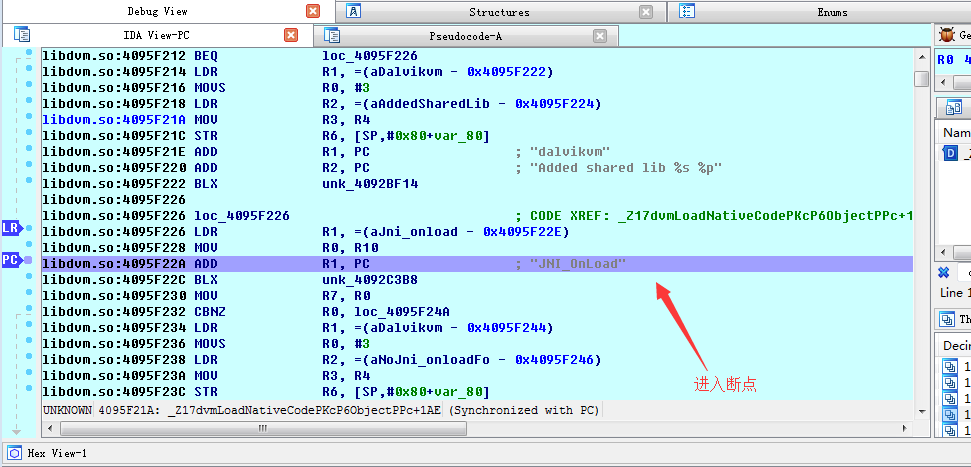
启动gba游戏

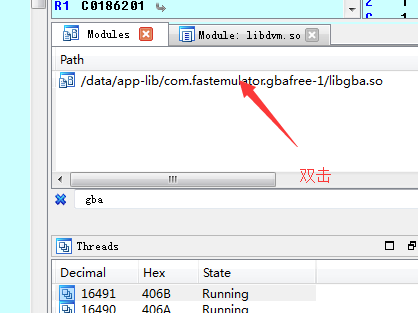


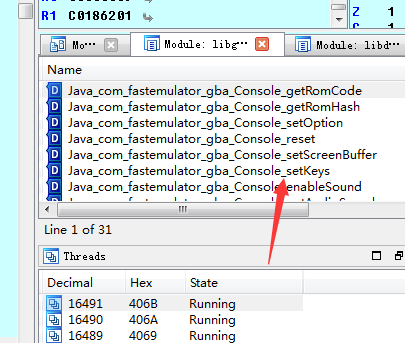
直接按F9

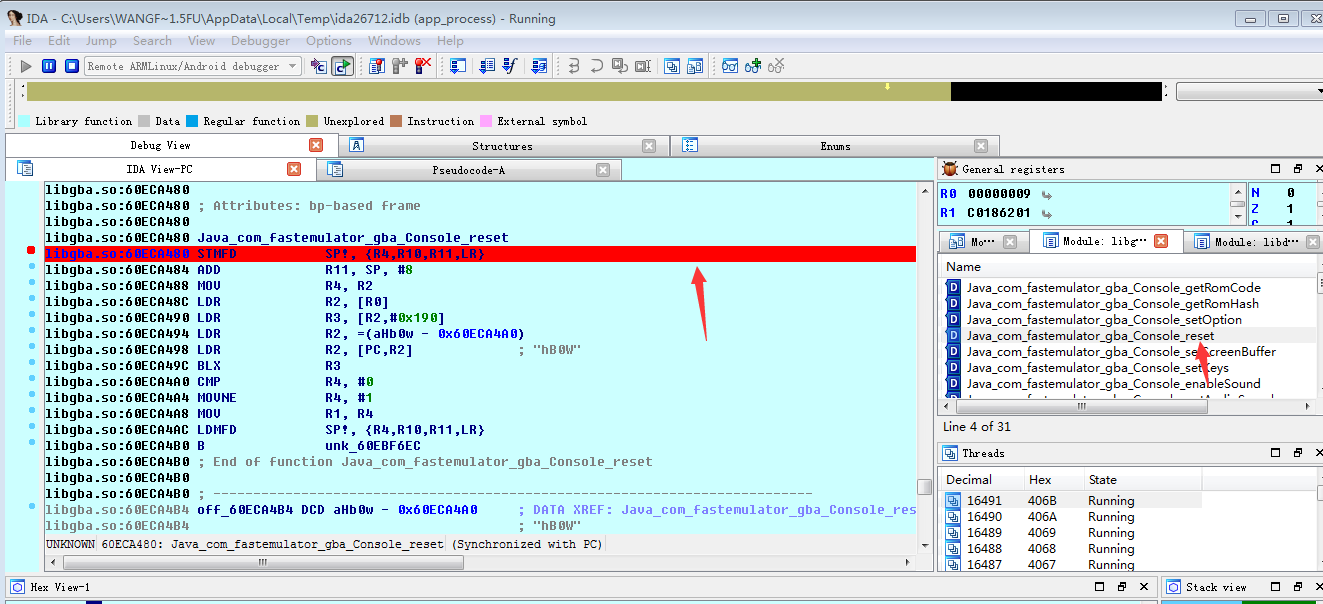


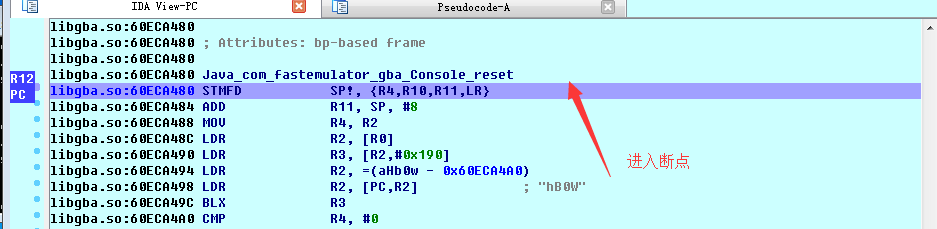




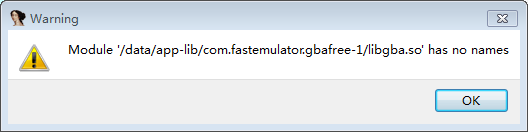








遇到的问题：



这样就无法在jni方法里面下断点了

解决方法：libdvm.so里面的dvmloadnativecode函数调用JNI\_Onload的地方设置断点。