我们的debug固件一般都是从nxp官网下载的。nxp.com/opensda。但有时候我们想要bootloader和固件的源码，根据自己要求进行修改。所以我们介绍一下daplink这个开源项目。Arm Mbed DAPLink是一个开源软件项目，可对运行在Arm Cortex CPU上的应用程序软件进行编程和调试。 DAPLink通常称为接口固件，它在连接到应用程序MCU的SWD或JTAG端口的辅助MCU上运行。

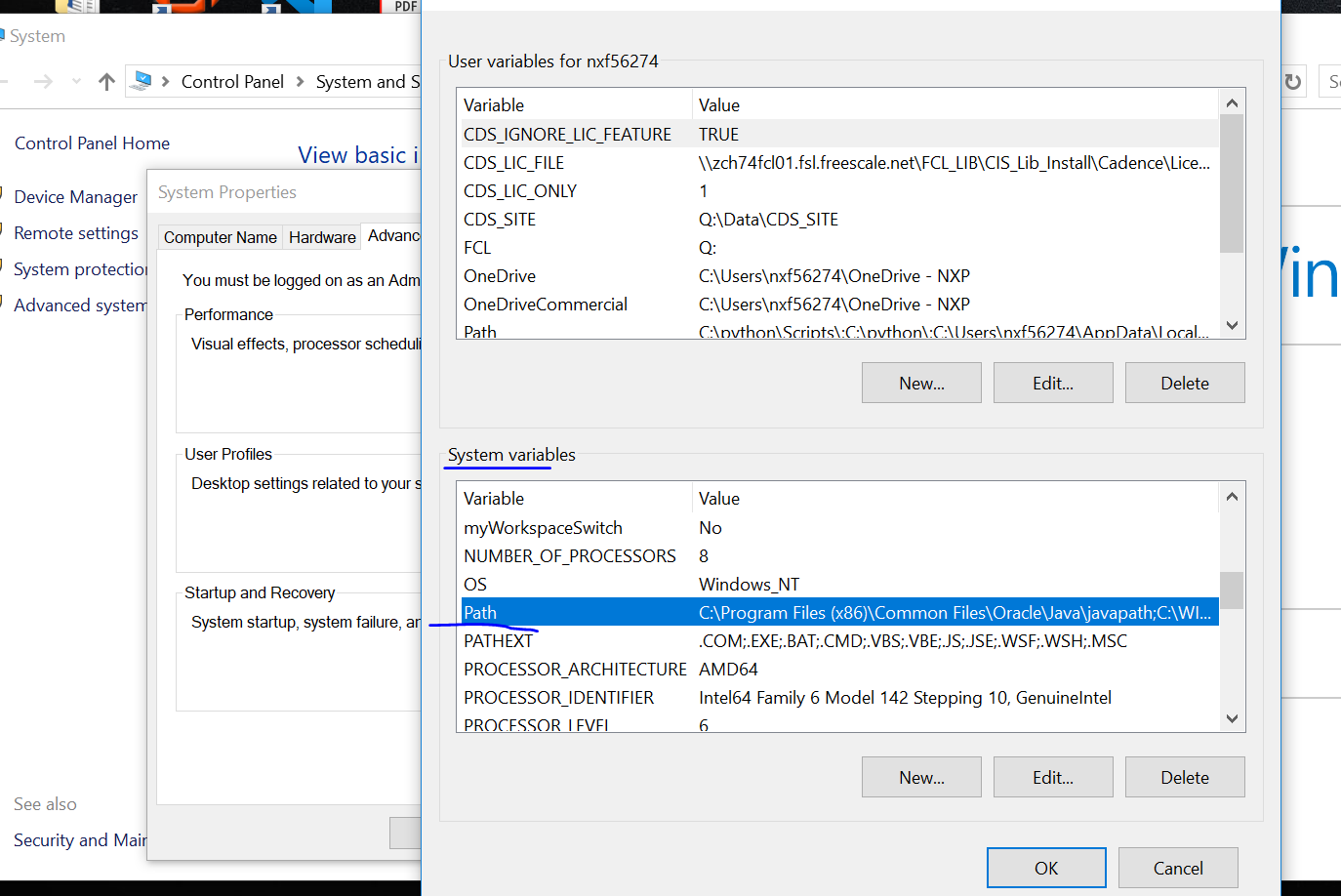
它提供了k20的bootloader和接口固件以及k26的bootloader和接口固件。很多frdm板子使用的是k20作为debugger，也有少数板子使用k26作为debugger。

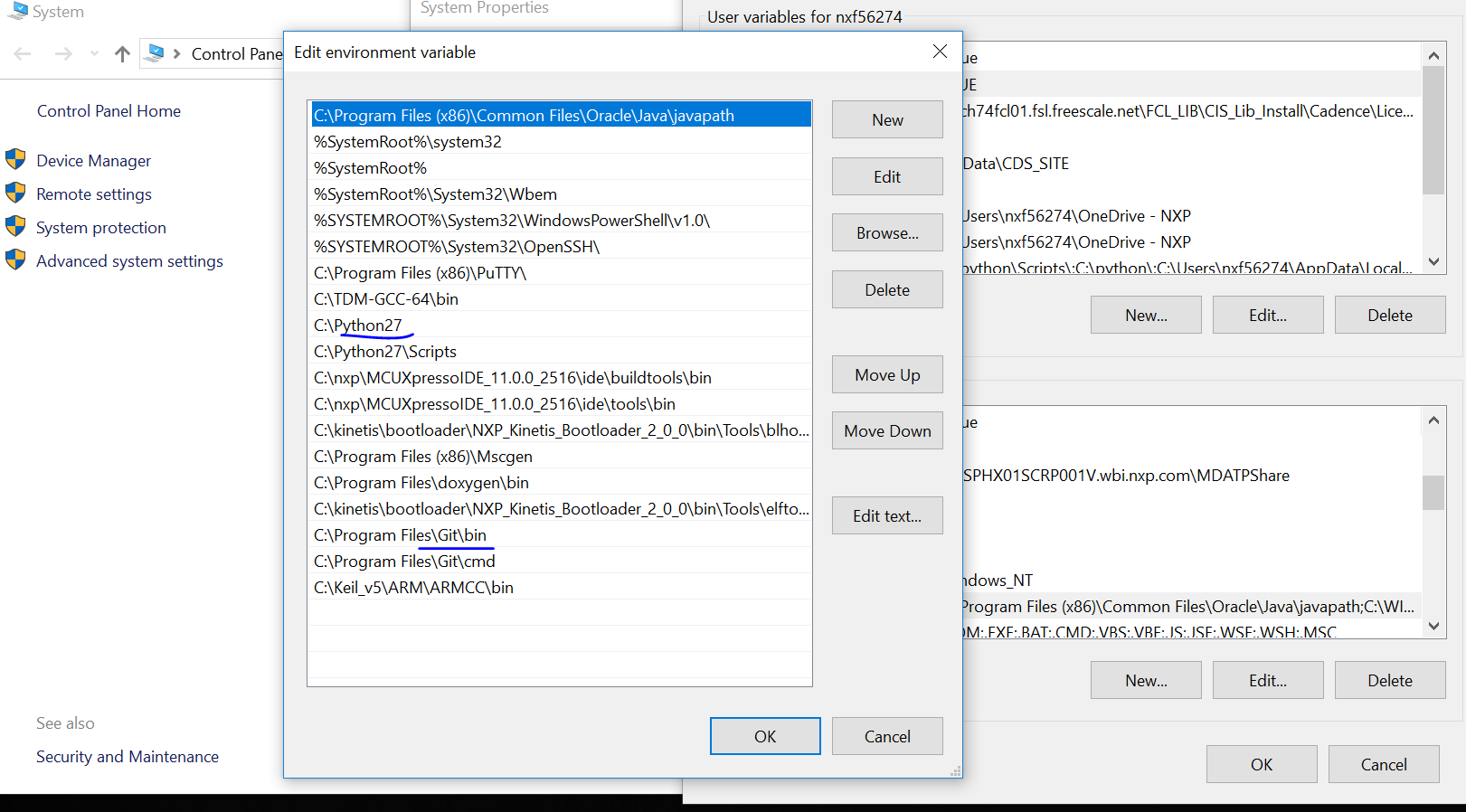
板子：FRDM-K64

电脑操作系统：WIN10

工程安装步骤：

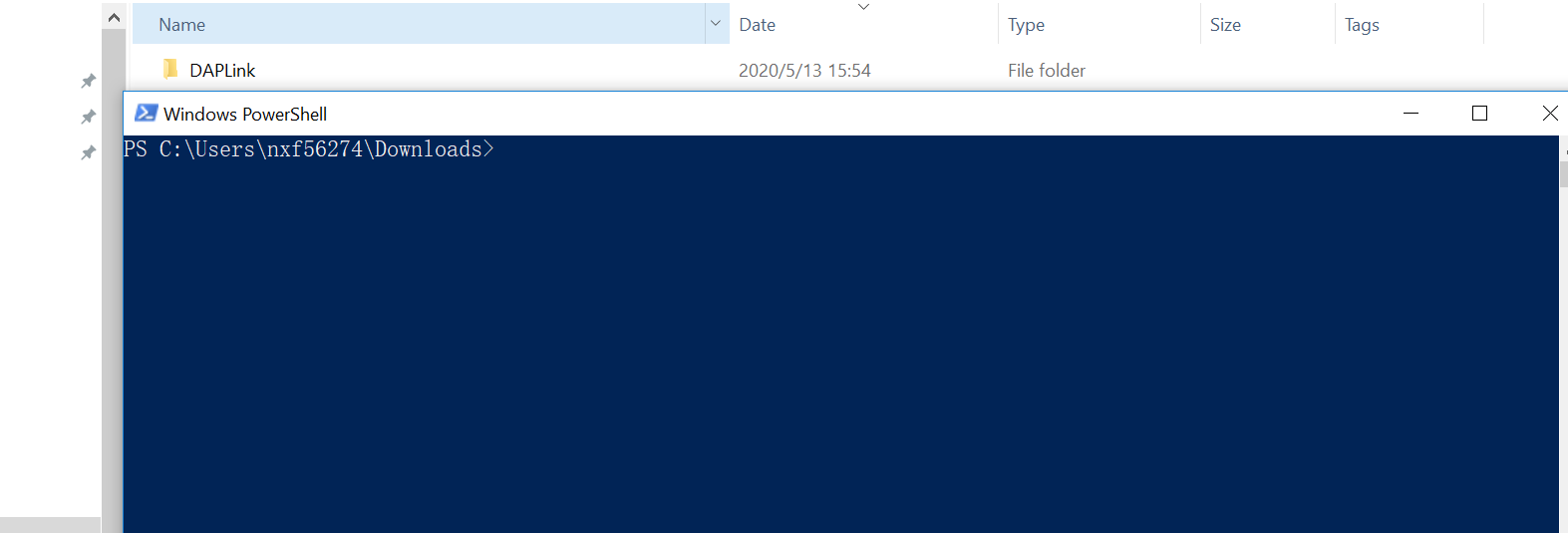
1. 安装git，python2.7.11以上，将这两个软件加入电脑系统环境变量（必须），最好将python下的scripts文件夹也加入环境变量，安装keil。DAPlink暂时只支持keil





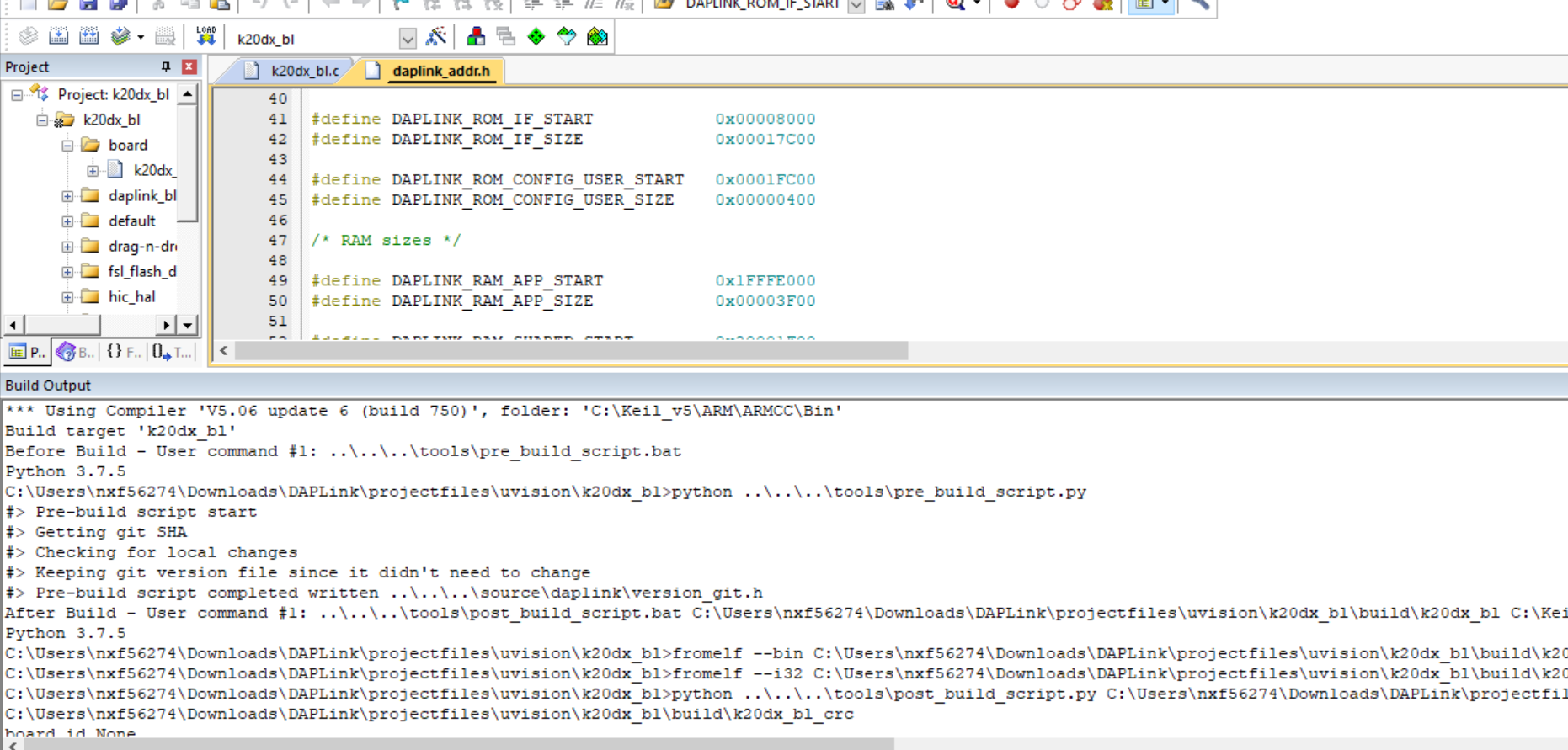
1. 使用python安装pip，可以网上搜教程
2. 安装virtualenv，使用powershell（按住shift，点击鼠标右键）, 输入pip install virtualenv
3. 之后命令都是在powershell下完成。获取源码

git clone <https://github.com/mbedmicro/DAPLink>，会在你当前目录生成一个DAPLink文件夹

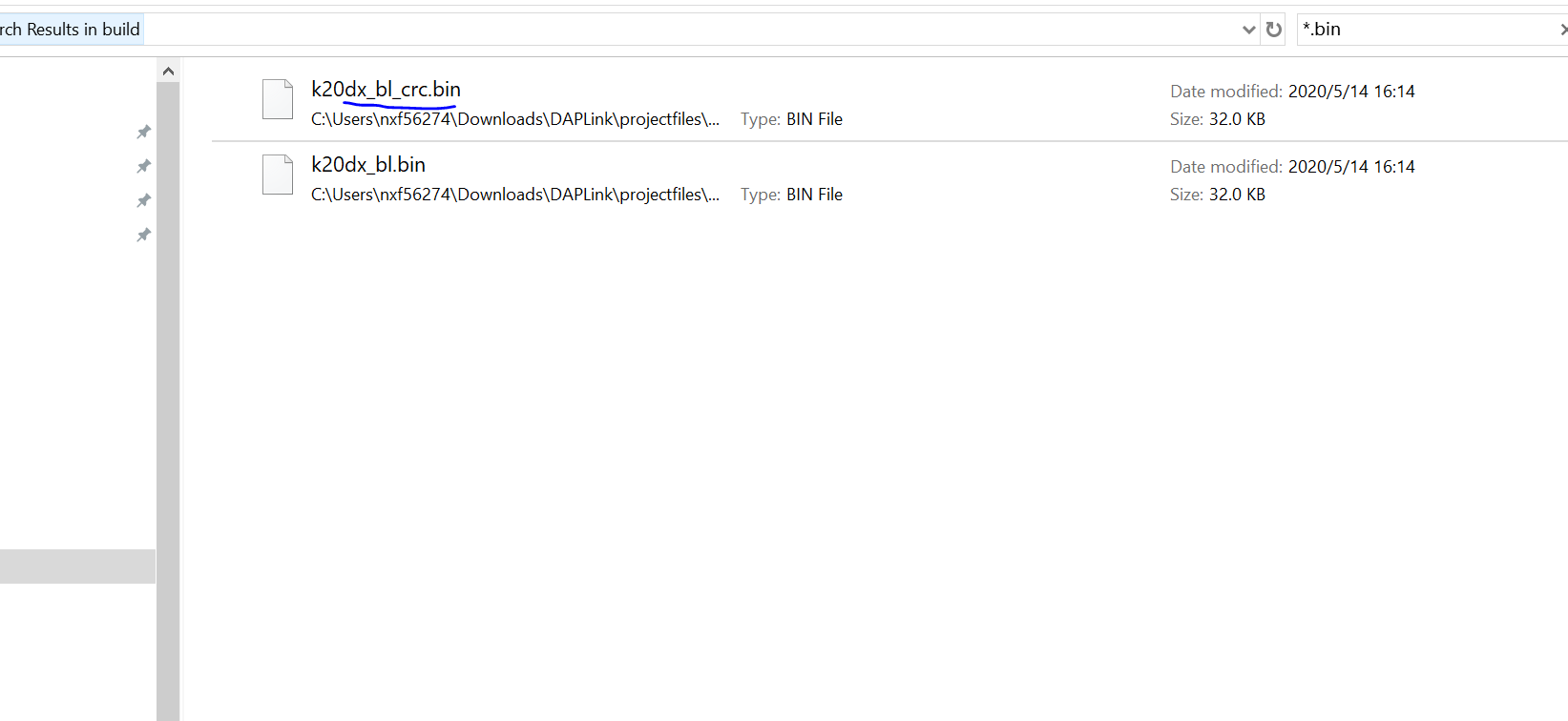


1. 进入目录cd DAPLink，这个文件夹下的docs/DEVELOPERS-GUIDE.md，是更加详细的如何使用这个DAPLink。
2. 安装virtualenv ，pip install virtualenv
3. 创建虚拟环境，virtualenv venv
4. 输入venv/Scripts/activate.bat激活虚拟环境
5. 安装必要工具，pip install -r requirements.txt
6. 产生keil工程，progen generate -t uvision，会产生projectfiles/uvision，进入文件夹就会发现各种bootloader和固件。名字带有bl的是bootloader，名字带有if是接口固件，要拖进mcu里的。打开第一个关于k20的工程，编译以后会产生bin文件，bin文件带crc的是我们要烧录的或者要拖的。对于名字if是一样的。编译时候会调用git命令，如果不添加这个命令到环境变量，编译就无法通过。

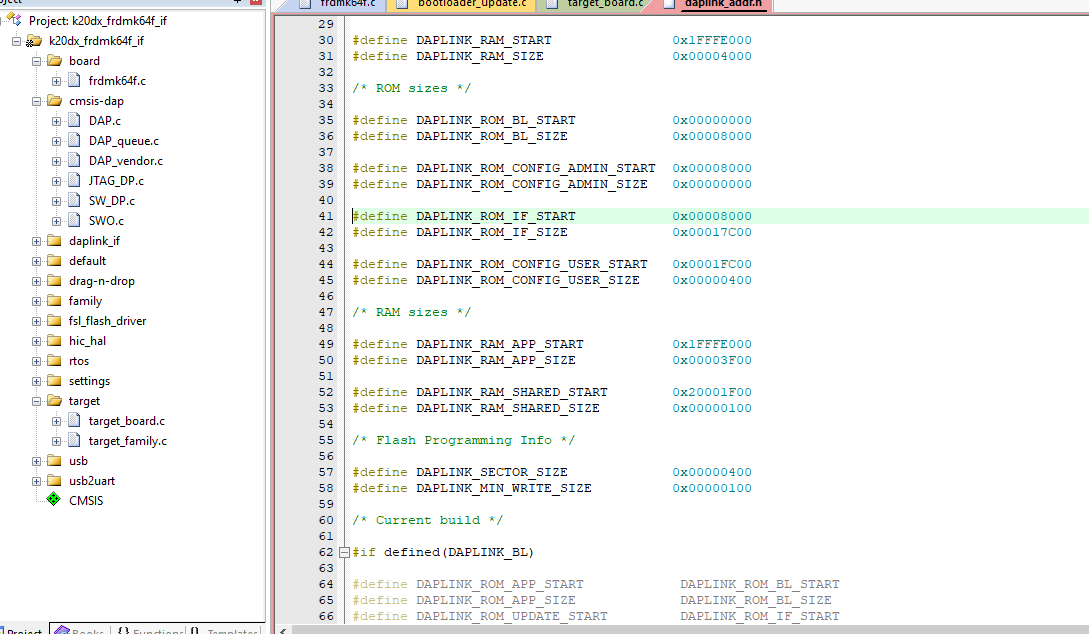
这是bootloader源码



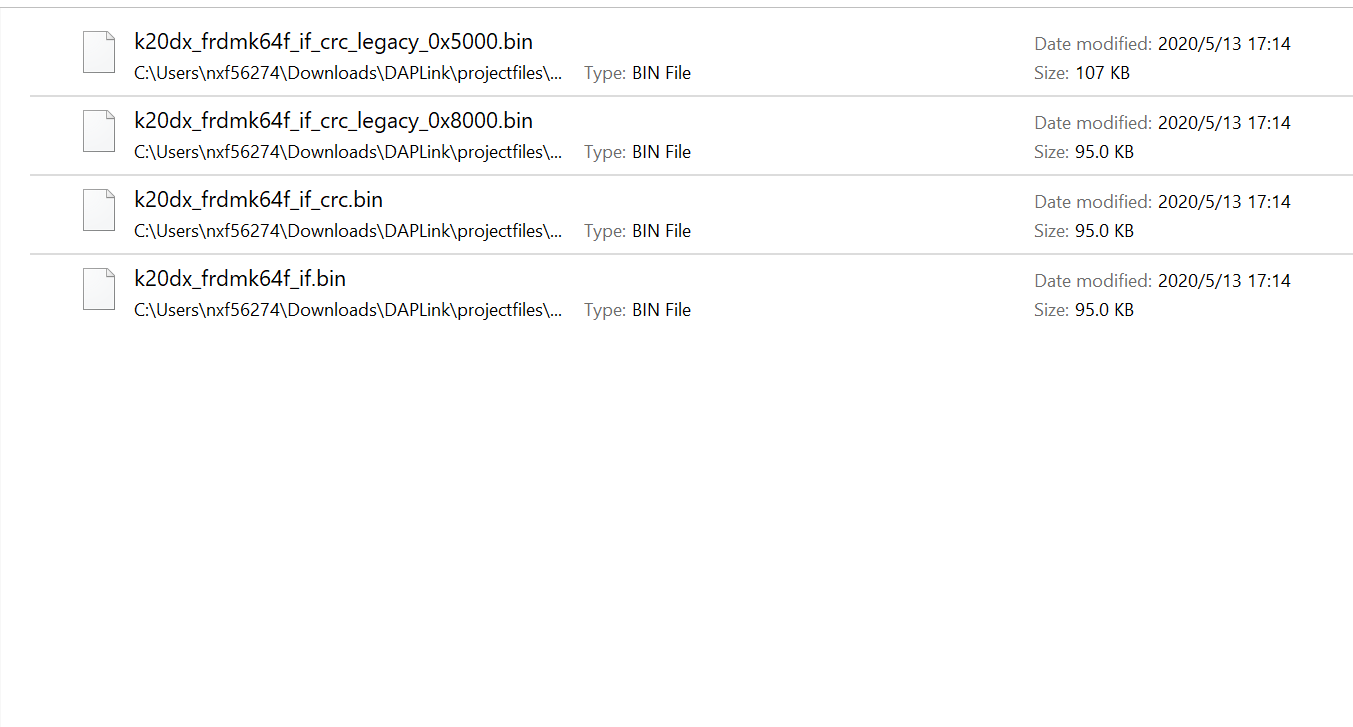
生成的bin文件



这是接口固件



生成的bin文件，带0x的就是不同的固件地址，一般DAPLink的bootloader默认的固件地址是0x8000，从上图可以看到是这个宏定义DAPLINK\_ROM\_IF\_START，所以我们要拖的文件就是带名字0x8000的文件，如果在bootloader里修改了固件起始地址，接口固件也要对应修改。



将bootloader烧入k20，再拖接口固件进入k20就可以看到这样结果

