Compile the firmware by yourself

Our debug firmware is generally downloaded from the official website of nxp. nxp.com/opensda. But sometimes we want to modify the source code of bootloader and firmware according to our own requirements. So we introduce the open source project daplink. Arm Mbed DAPLink is an open source software project that can program and debug application software running on the Arm Cortex CPU. DAPLink is usually called interface firmware, and it runs on the auxiliary MCU connected to the SWD or JTAG port of the application MCU.

It provides k20 bootloader and interface firmware and k26 bootloader and interface firmware. Many frdm boards use k20 as a debugger, and a few boards use k26 as a debugger.
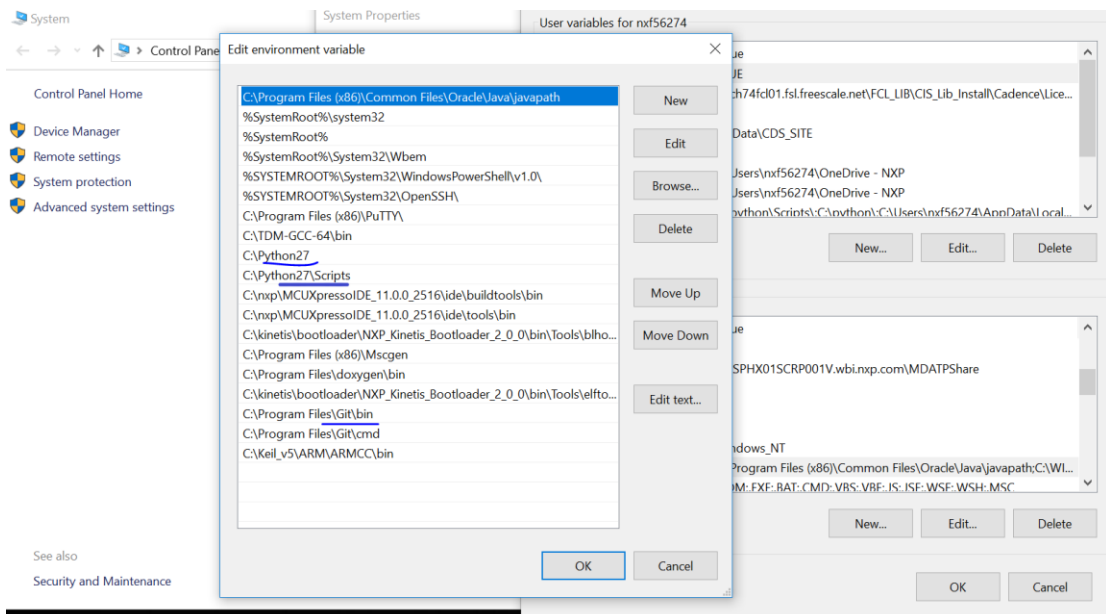
board：FRDM-K64

OS：WIN10

steps：

1. Install git, python2.7.11 or above, add these two software to the computer system environment variables (required), it is best to add the scripts folder under python to the environment variables, and install keil.   DAPlink currently only supports IDE keil.
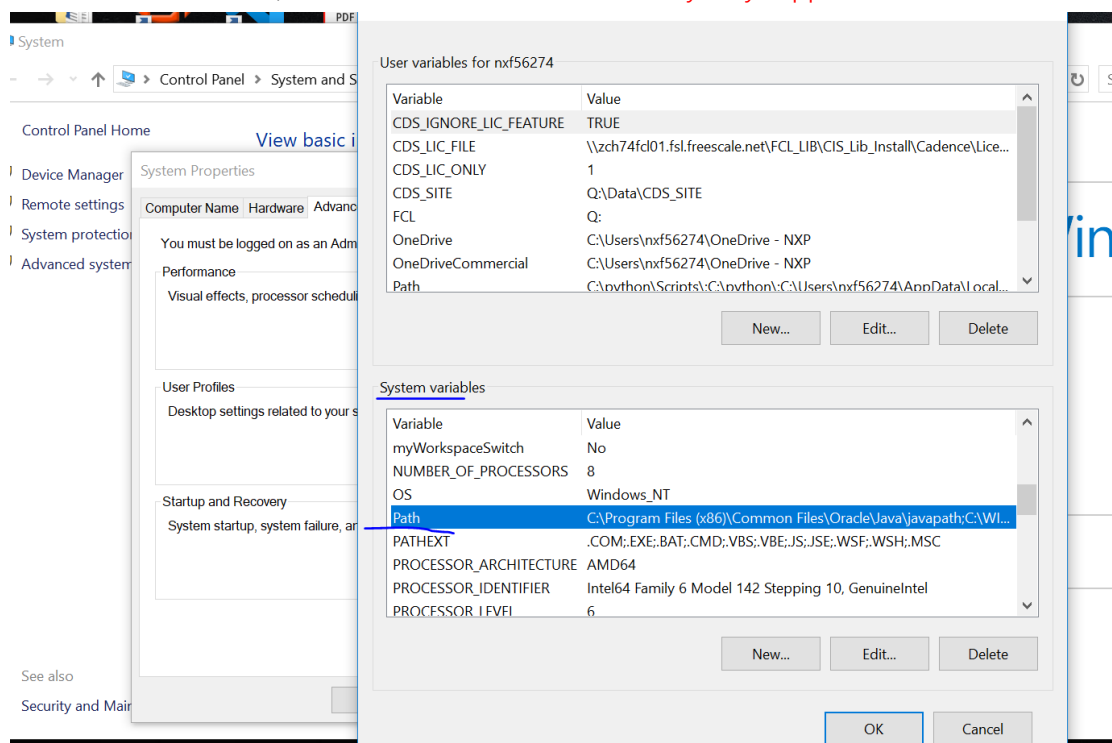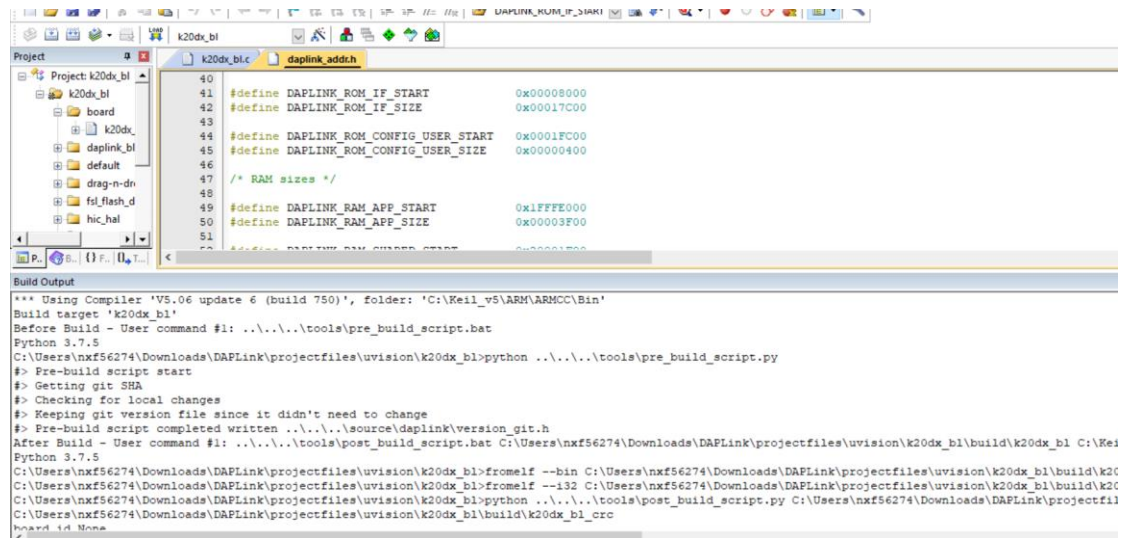
2. Use python to install pip, you can search for tutorials online

3. Install virtualenv, use powershell (hold down shift and click the right mouse button),

Input 'pip install virtualenv'

4. After that, the commands are all completed under powershell. Get the source code.

Input 'git clone https://github.com/mbedmicro/DAPLink'

Note: You must use git to download the code, or you will fail at compiling the code.

It Will generate a DAPLink folder in your current directory
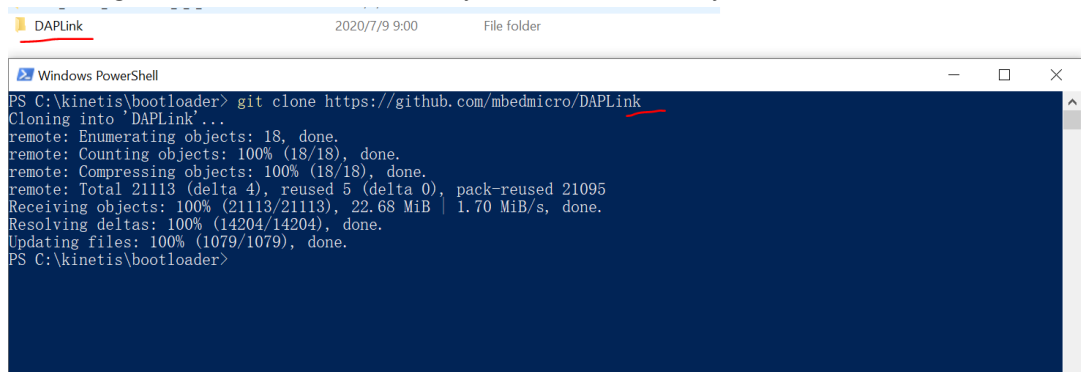
5. Enter the directory. Input 'cd DAPLink', The docs/DEVELOPERS-GUIDE.md under this folder is more detailed how to use this DAPLink

6. Create a virtual environment, Input 'virtualenv venv'

7. Input 'venv/Scripts/activate.bat' to active the virtual environment

8. Install necessary tools, 'pip install -r requirements.txt'

9. Generate keil project,

input 'progen generate -t uvision'

It will generate projectfiles/uvision, enter the folder and you will find various bootloader and firmware. The name with 'bl' is the bootloader, and the name with 'if' is the interface firmware, which is to be dragged into the mcu. Open the first project about k20. After compilation, a bin file will be generated. The bin file with crc is what we want to burn or drag. For the name 'if' is the same. The git command will be called during compilation. If you do not add this command to the environment variable, the compilation will fail.

This is the bootloader source code

Bin file

This is interface firmware.
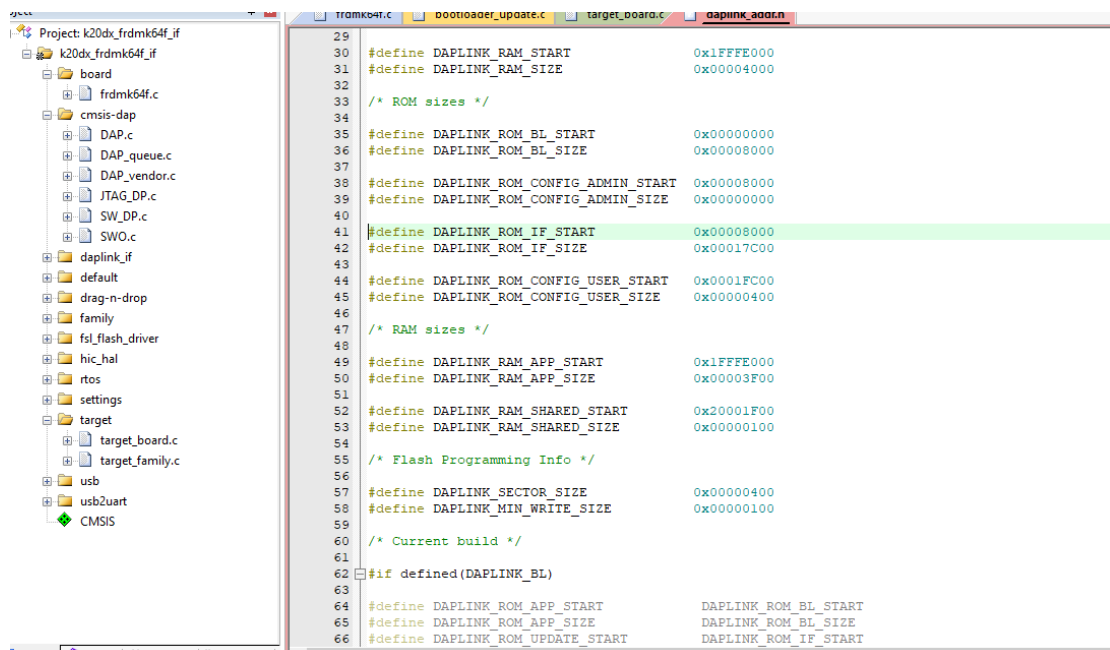
The generated bin file with '0x' is firmware address. Generally, the default firmware address of the DAPLink bootloader is 0x8000. As you can see from the above figure, this macro defines DAPLINK_ROM_IF_START, so the file we want to drag is the file with the name '0x8000'. If the firmware start address is modified in the bootloader, the interface firmware should also be modified accordingly

Burn the bootloader into k20, then drag the interface firmware into k20 to see this result