

使用clion搭配stm32

在这个框架下开发环境需要：

- Clion : IDE
- CubeMX : 代码生成器
- arm-gcc : 工具链
- openocd : 固件烧录软件

1 首先是官方手册

官方手册的地址 [STM32CubeMX.projects | CLion Documentation \(jetbrains.com\)](#)

2 环境准备

2.1 下载安装Clion

进入官网下载Clion

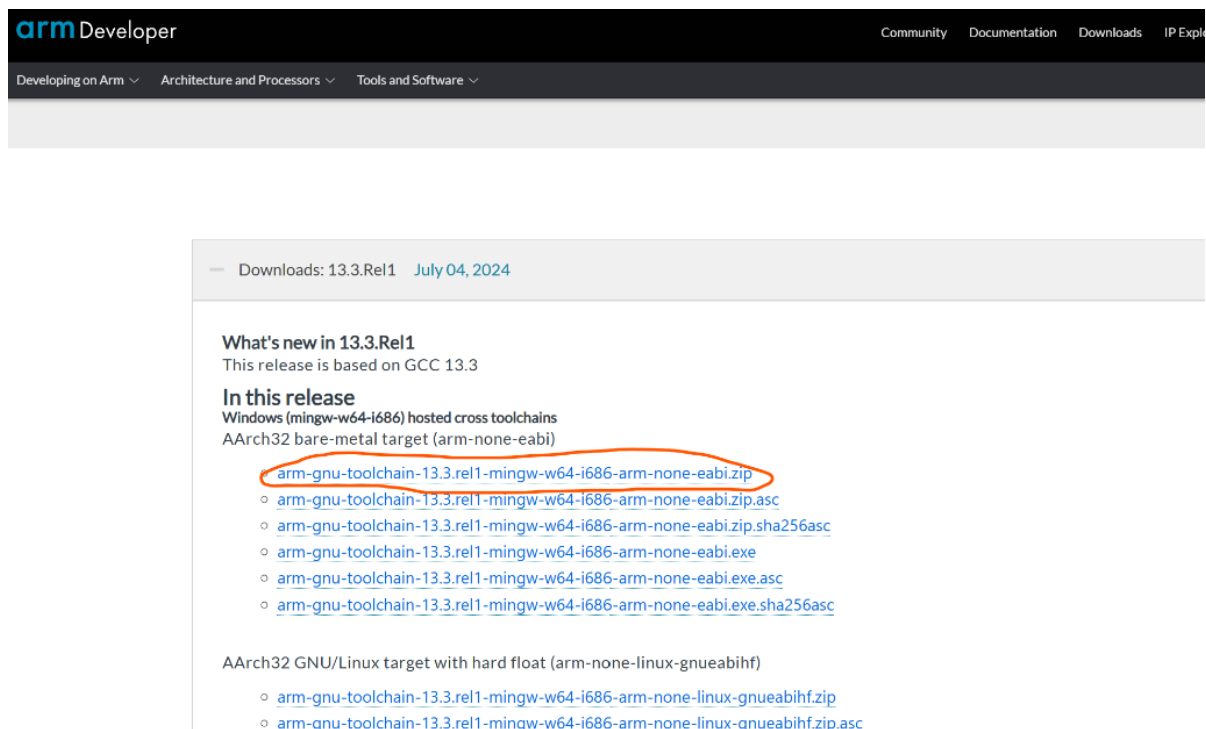
[Download CLion: A Smart Cross-Platform IDE for C and C++ \(jetbrains.com\)](#)

选择安装路径的时候，安装路径不要有空格和中文

安装选项给他全勾上

2.2 下载arm-gcc

- 进入官网下载windows版的arm交叉编译器[Arm GNU Toolchain Downloads – Arm Developer](#)



- 下载完后，解压缩，我是把解压缩完的文件放在D盘，看个人喜好了
- 然后进入解压的bin文件夹，把这个路径添加到系统环境变量PATH中
- 然后在命令行里面确认添加成功没有，打开cmd或者powershell都行

```
C:\Users\35142> arm-none-eabi-gcc -v
Using built-in specs.
COLLECT_GCC=D:\msys64\mingw64\bin\arm-none-eabi-gcc.exe
COLLECT_LTO_WRAPPER=D:/msys64/mingw64/bin/./lib/gcc/arm-none-eabi/13.3.0/lto-wrapper.exe
Target: arm-none-eabi
Configured with: ../gcc-13.3.0/configure --prefix=/mingw64 --target=arm-none-eabi --with-native-system-header-dir=/mingw64/include --libexecdir=/mingw64/lib --enable-languages=c,c++ --enable-plugins --disable-decimal-float --disable-libffi --disable-libgomp --disable-libmudflap --disable-libquadmath --disable-libssp --disable-libstdc++-pch --disable-nls --disable-shared --disable-threads --disable-tls --disable-libada --with-gnu-as --with-gnu-ld --with-system-zlib --with-newlib --with-headers=/mingw64/arm-none-eabi/include --with-python-dir=share/gcc-arm-none-eabi --with-gmp --with-mpfr --with-mpc --with-isl --with-libelf --enable-gnu-indirect-function --with-multilib-list=rmprofile --with-host-libstdc++=-static-libgcc -Wl,-Bstatic,-lstdc++,-Bdynamic -lm' --enable-linker-plugin --plugin-flags='LDFLAGS=-static-libstdc++\ -static-libgcc\ -Wl,--stack,12582912' LDFLAGS=' -Wl,--disable-dynamicbase'
Thread model: single
Supported LTO compression algorithms: zlib zstd
gcc version 13.3.0 (GCC)
```

2.3 下载openocd

- 先进入第三方网站下载openocd[Download OpenOCD for Windows \(gnutoolchains.com\)](https://gnutoolchains.com/)

[PRODUCTS](#)
[DOWNLOADS](#)
[TUTORIALS](#)
[SUPPORT](#)
[CONTACT](#)

Toolchains

- Android (GDB only)
- ARM
- AVR
- Beaglebone
- Blackfin
- Cubieboard
- ESP32
- ESP8266
- Nvidia Jetson
- Kendryte
- m32c-elf
- m32r-elf
- m68k-elf
- MinGW32
- MinGW64
- MSP430
- PowerPC

Download pre-built OpenOCD for Windows

OpenOCD is an open-source tool that allows debugging various ARM devices with GDB using a wide variety of JTAG programmers. You can download the pre-built OpenOCD for windows from this page:

Version	Download link
20240904	openocd-20240904.7z
20240820	openocd-20240820.7z
20240813	openocd-20240813.7z
20231002	openocd-20231002.7z
20230712	openocd-20230712.7z
20230621	openocd-20230621.7z
20230202	openocd-20230202.7z

Each build above includes the necessary binaries and scripts to begin debugging your device right away.

- 同样解压缩完，添加bin路径到PATH里面

D:\msys64\mingw64\arm-none-eabi\bin

D:\openocd-20240904\OpenOCD-20240904-0.12.0\bin

2.4 下载CubeMX

进入官网下载

[STM32CubeMX - STM32Cube初始化代码生成器 - 意法半导体STMicroelectronics](#)

2.5 安装驱动

stlink驱动

目前个人用的还是stlink，后面应该会更换成jlink

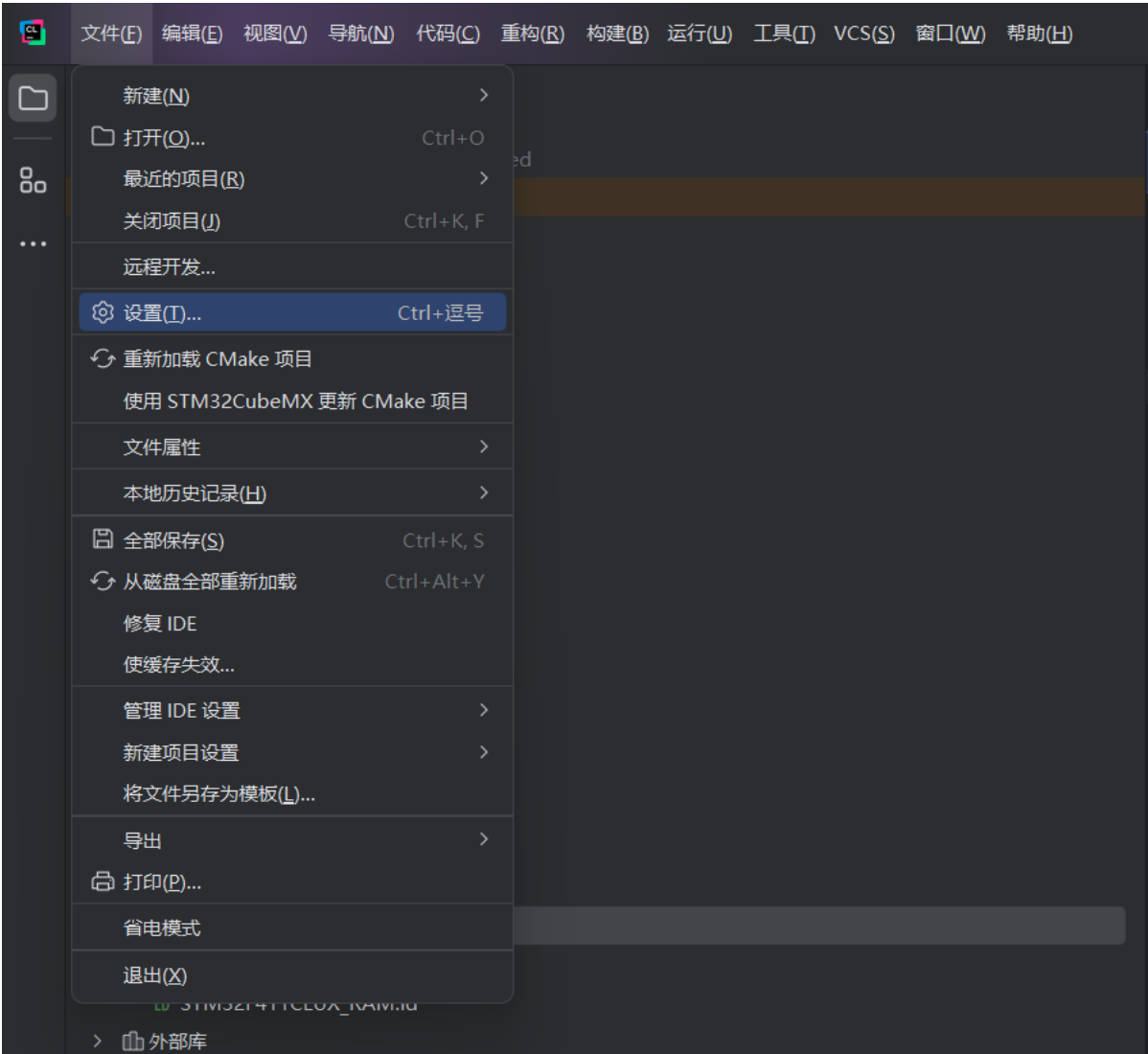
jlink驱动

3 开始配置Clion

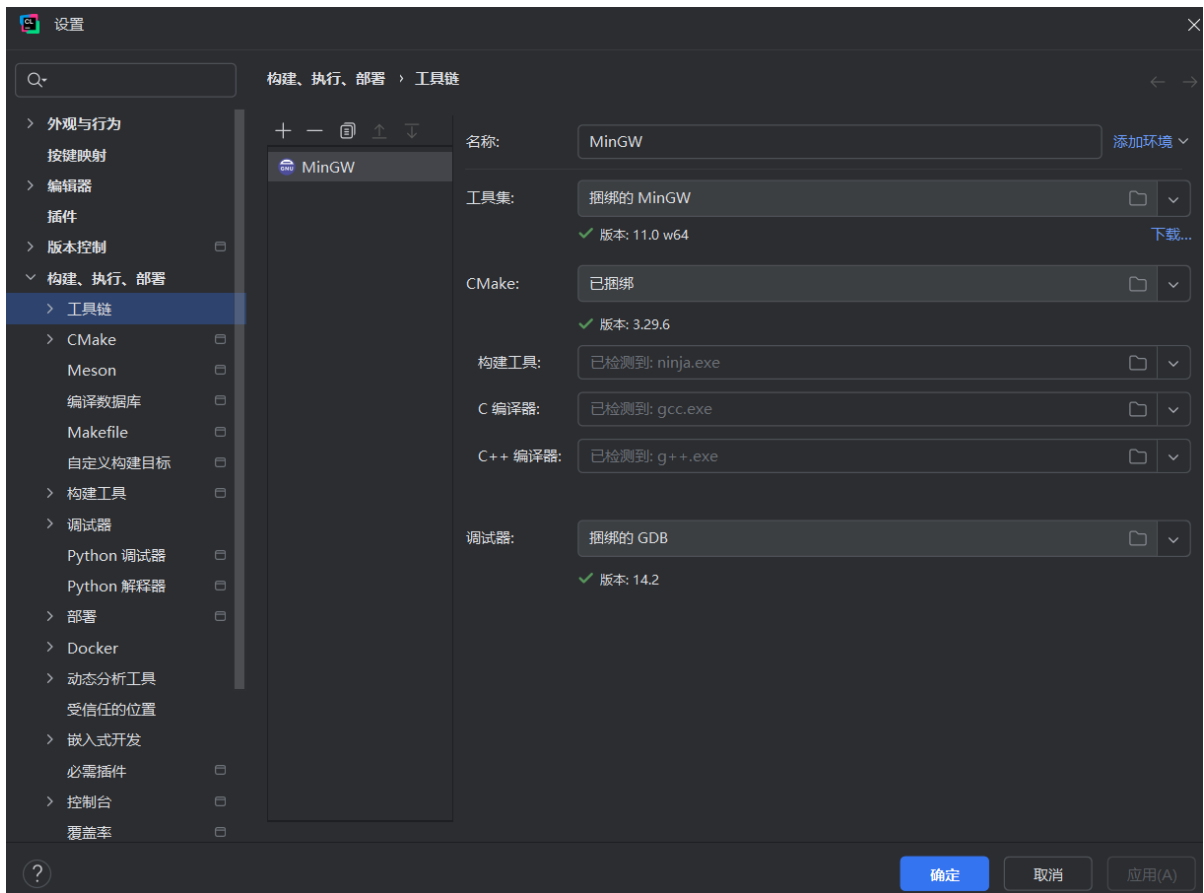
打开Clion，破解的话，打开clion.zip那个文件夹，解压缩完看文件夹里面的说明

3.1 路径设置

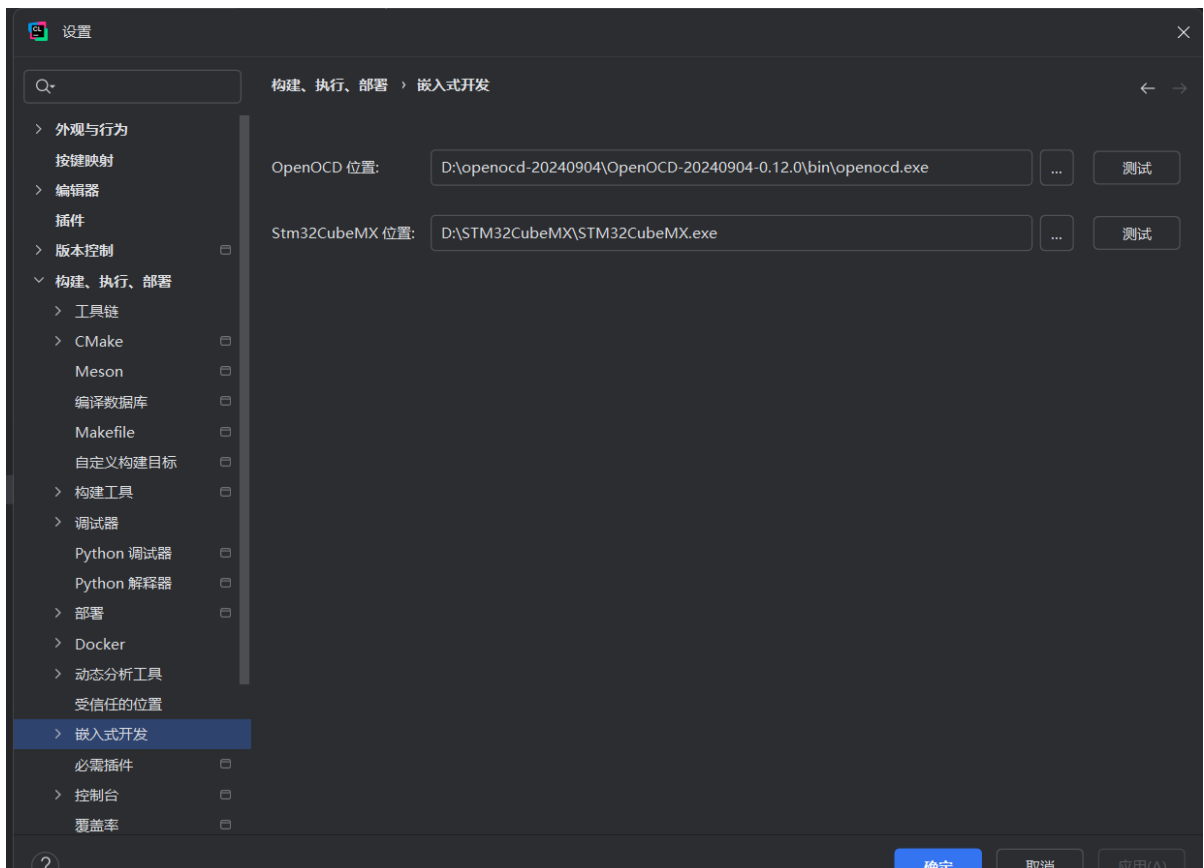
- 打开设置



- 然后先检查工具链

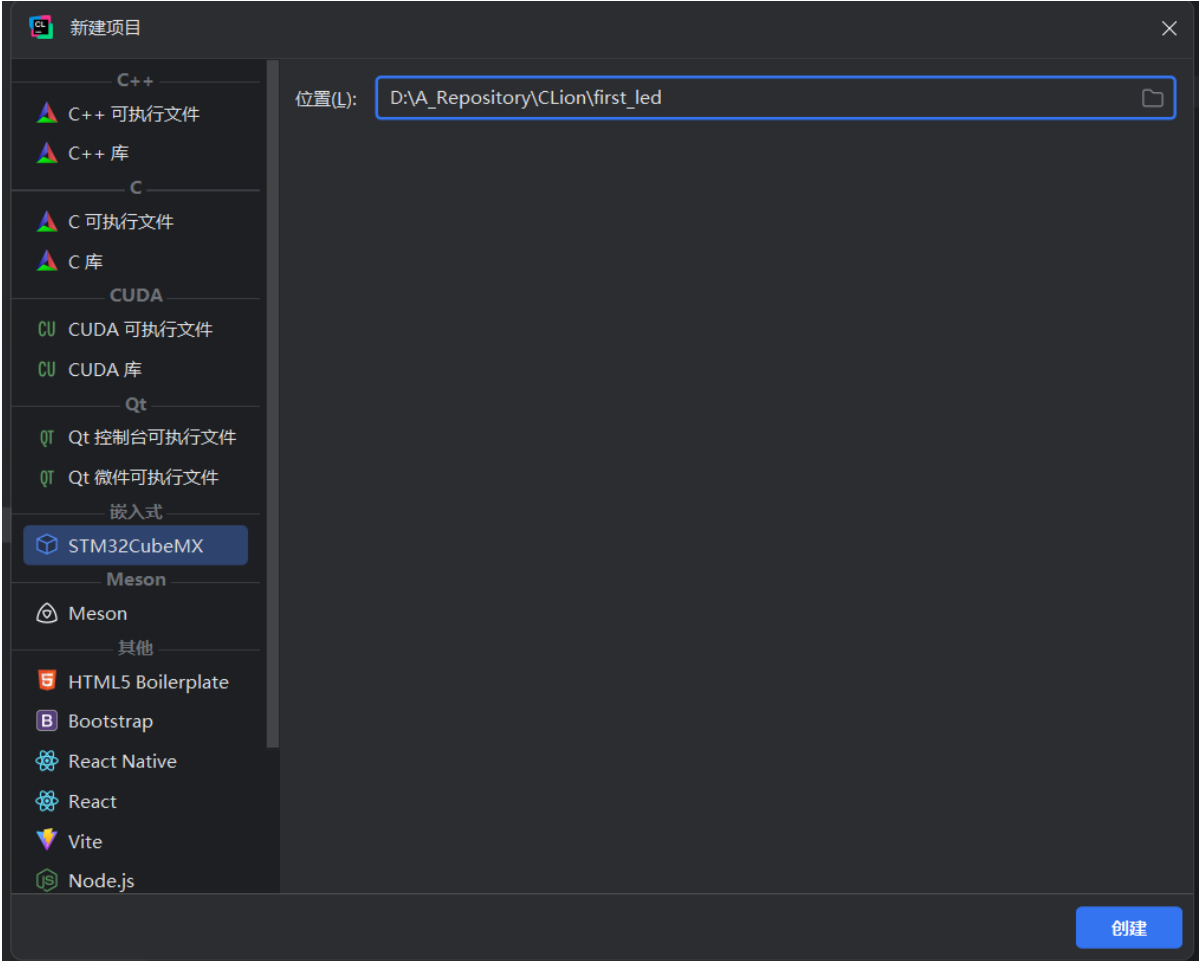


- 然后再在嵌入式开发里面，检查openocd和cubemx的位置，换成你自己下载这两个东西的地址

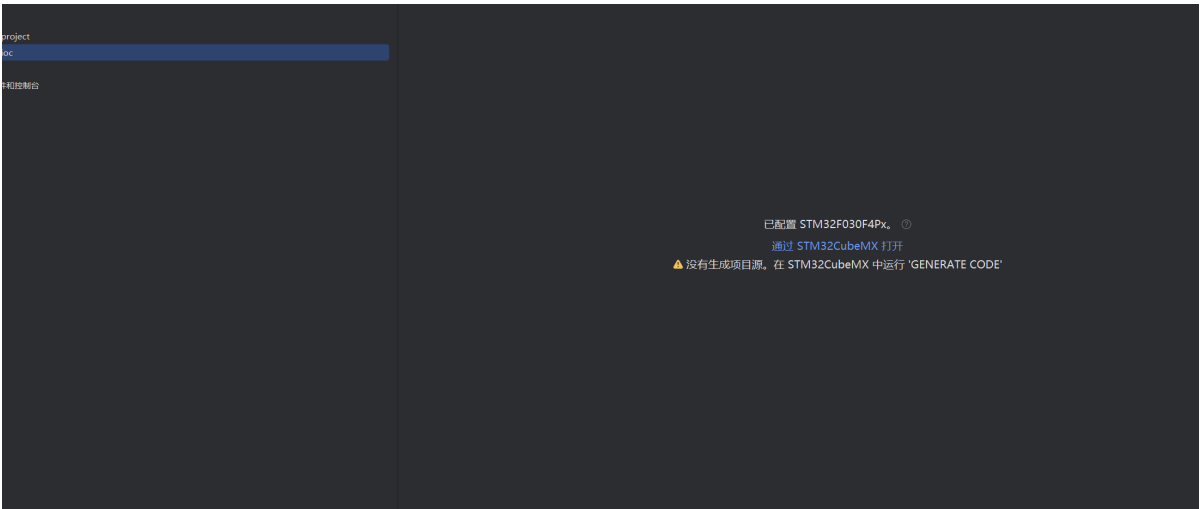


3.2 创建新的cubemx工程

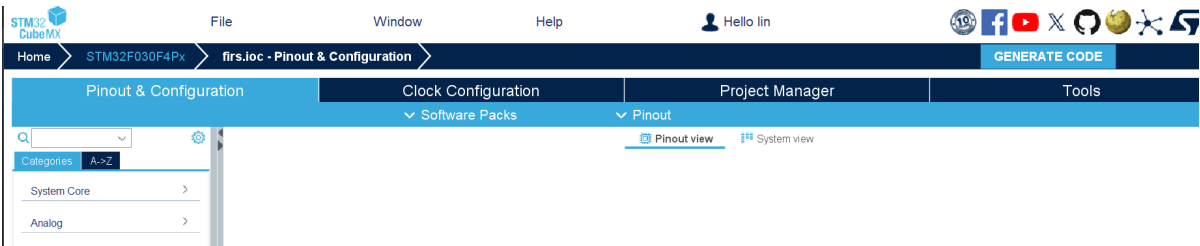
- 创建路径根据个人爱好习惯



- 点击旁边的.ioc文件，再通过STM32CubeMX打开

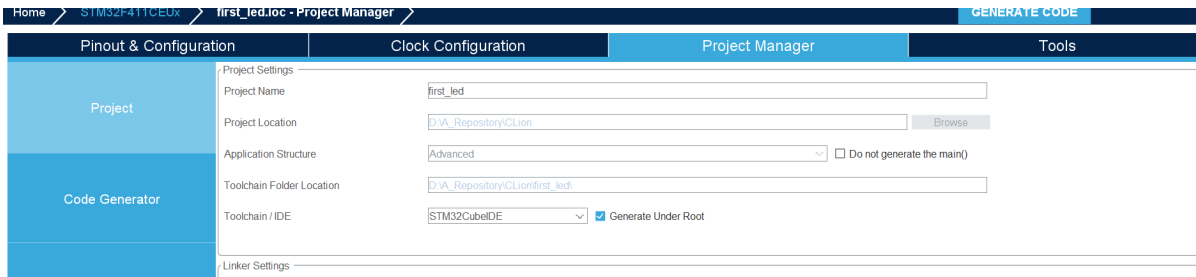


- 点击左上角的STM32F030F4Px，更换自己需要的芯片



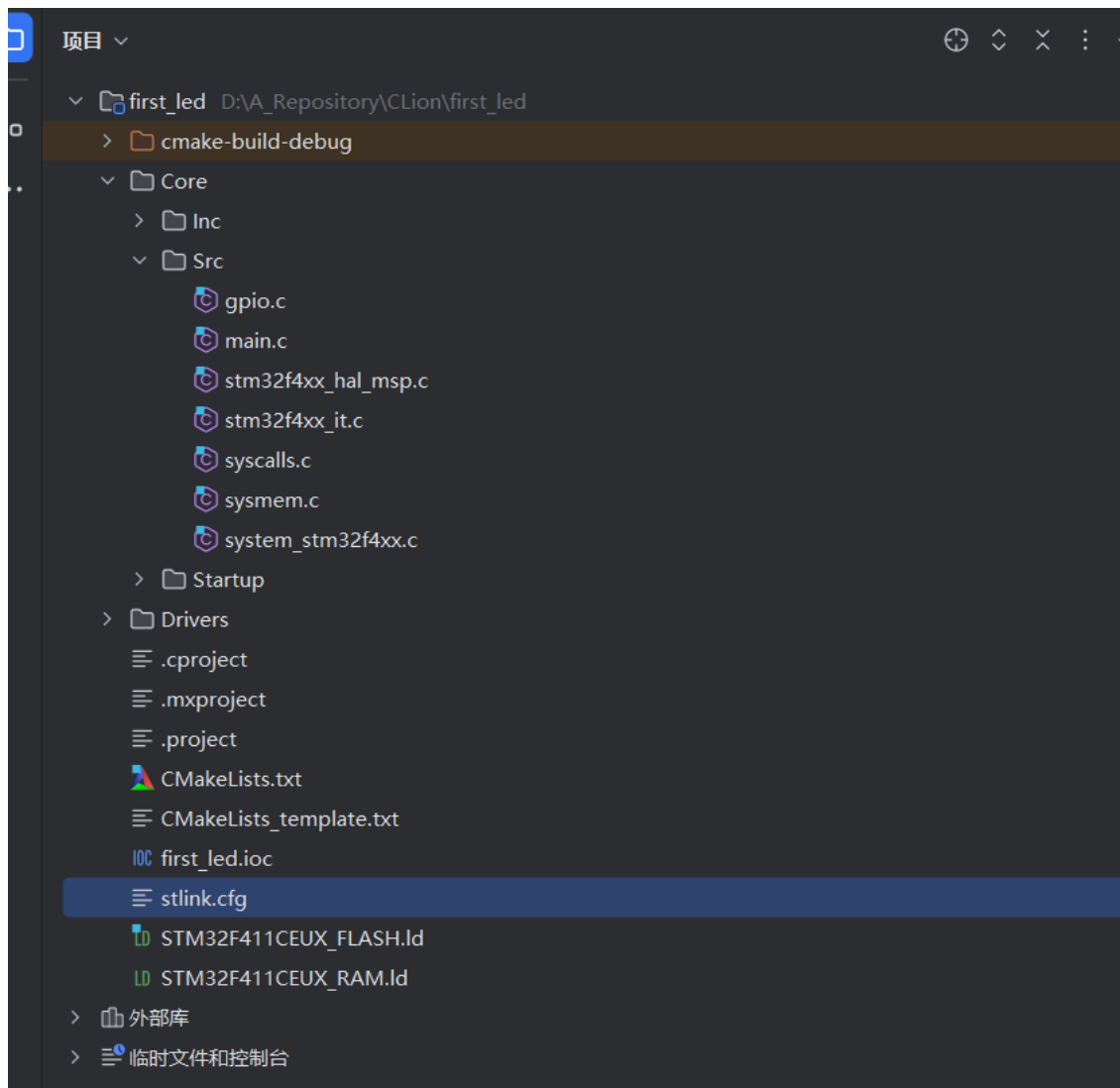
其余的就按照正常的cubemx去配置就好了

需要注意的只有这里，跟别的是不一样的



3.3 烧录文件配置

- 在工程里面加入stlink.cfg文件



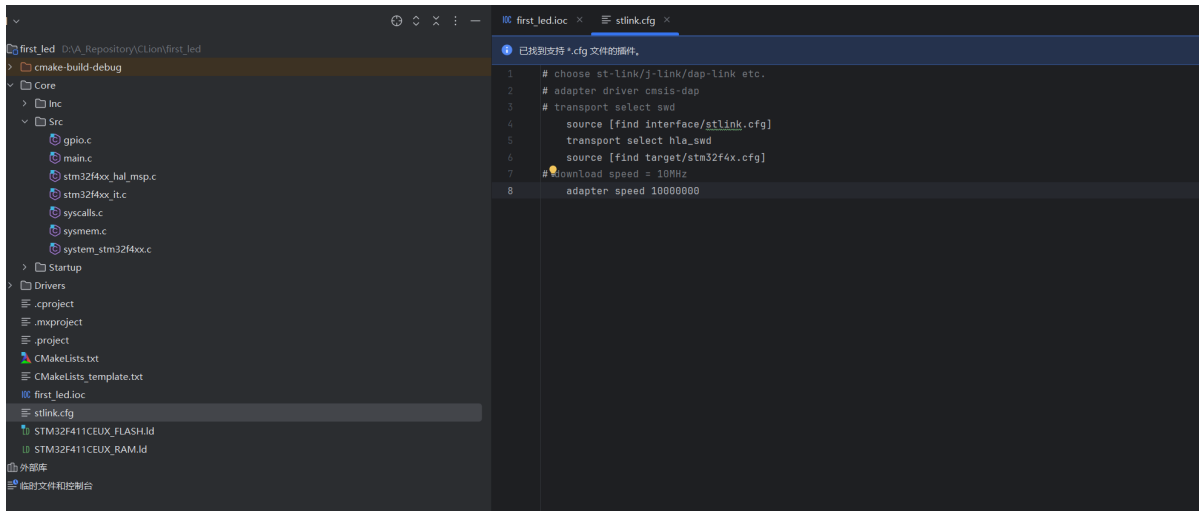
- 然后在里面加入配置代码：

很明显这个代码是给stlink、f4的芯片用的

```

1 # choose st-link/j-link/dap-link etc.
2 # adapter driver cmsis-dap
3 # transport select swd
4     source [find interface/stlink.cfg]
5     transport select hla_swd
6     source [find target/stm32f4x.cfg]
7 # download speed = 10MHz
8     adapter speed 10000000

```



如果后续要更改为daplink或者jlink，又或者要换芯片，就要对这个代码进行修改

解释下这个代码

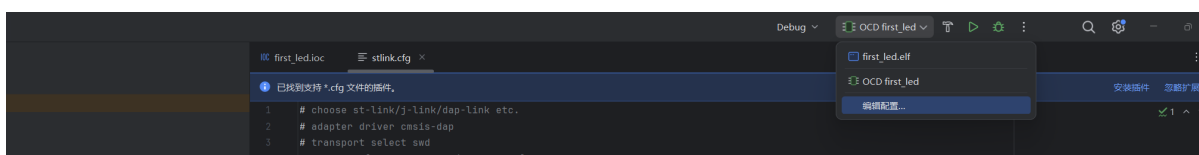
```

1 #设置使用的调试烧录器为ST-Linkv
2 source [find interface/stlink.cfg]
3
4 #设置使用的调试烧录器为J-link
5 source [find interface/jlink.cfg]
6
7 #设置使用的调试烧录器为DAP-Link
8 source [find interface/cmsis-dap.cfg]
9
10 # 调试接口选择swd，此类型是ST-Link专用的
11 transport select hla_swd
12
13 # 调试接口选择swd，此类型是J-link，DAP-Link通用
14 transport select swd
15
16 #设置目标芯片，选择stm32f1x，如果是f4系列的换成f4就行了其他系列以此类推
17 source [find target/stm32f1x.cfg]

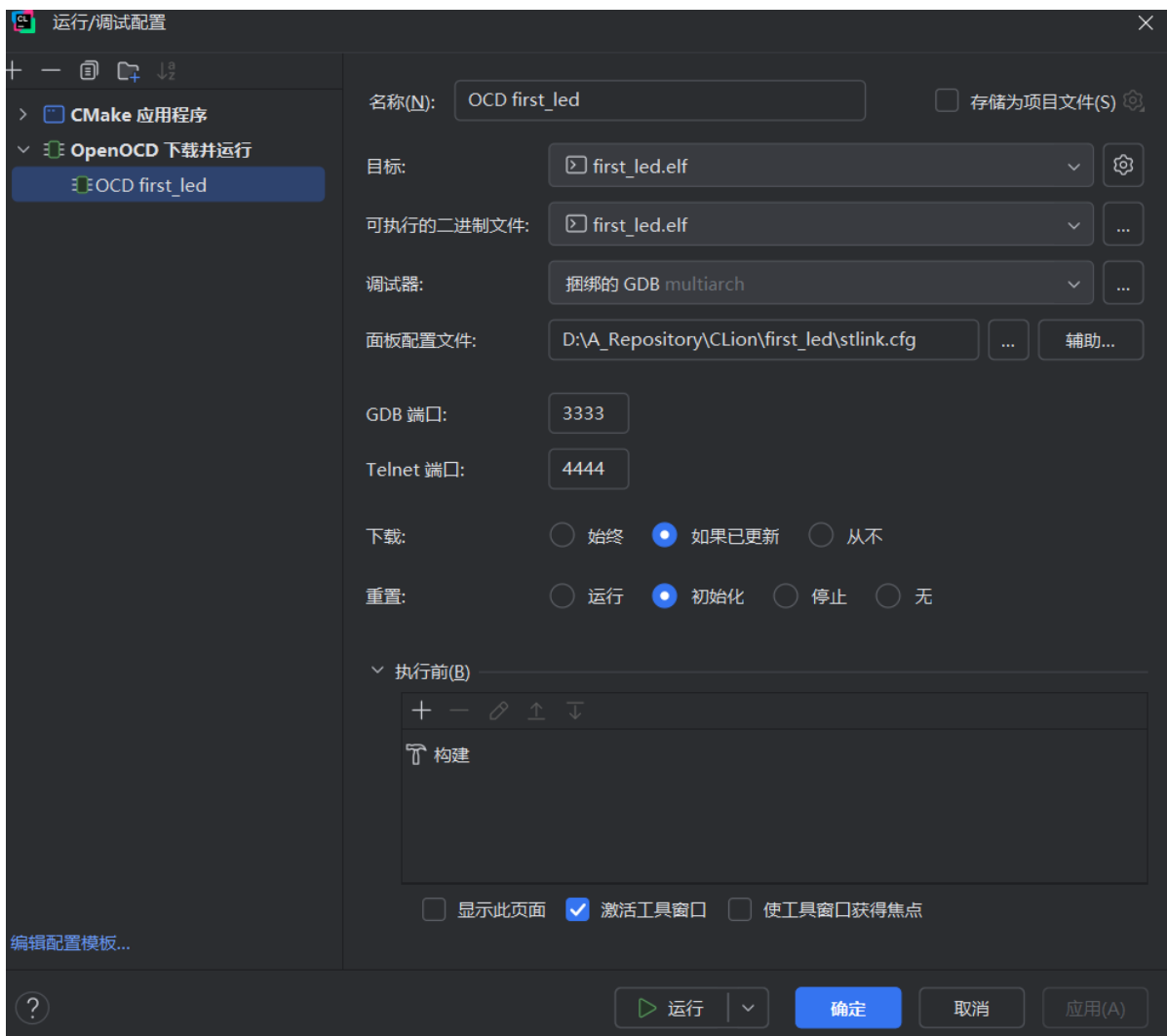
```

- 然后配置烧录

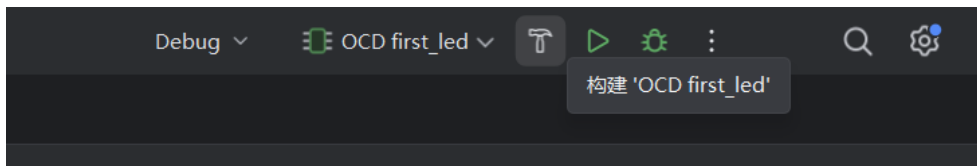
点击编辑配置



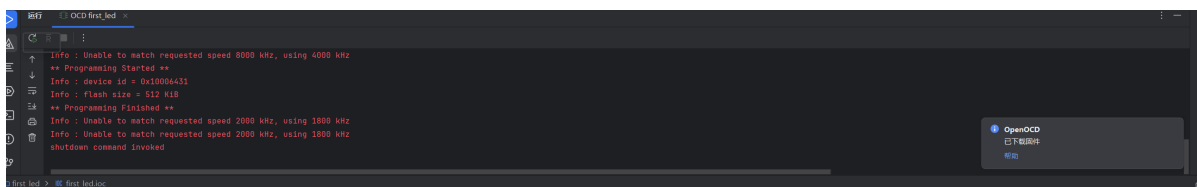
然后在i面板配置文件里面，把路径改成自己的stlink.cfg文件所在路径



3.4 构建工程，烧录



左上角那个小锤子是构建，然后右边那个三角形是烧录

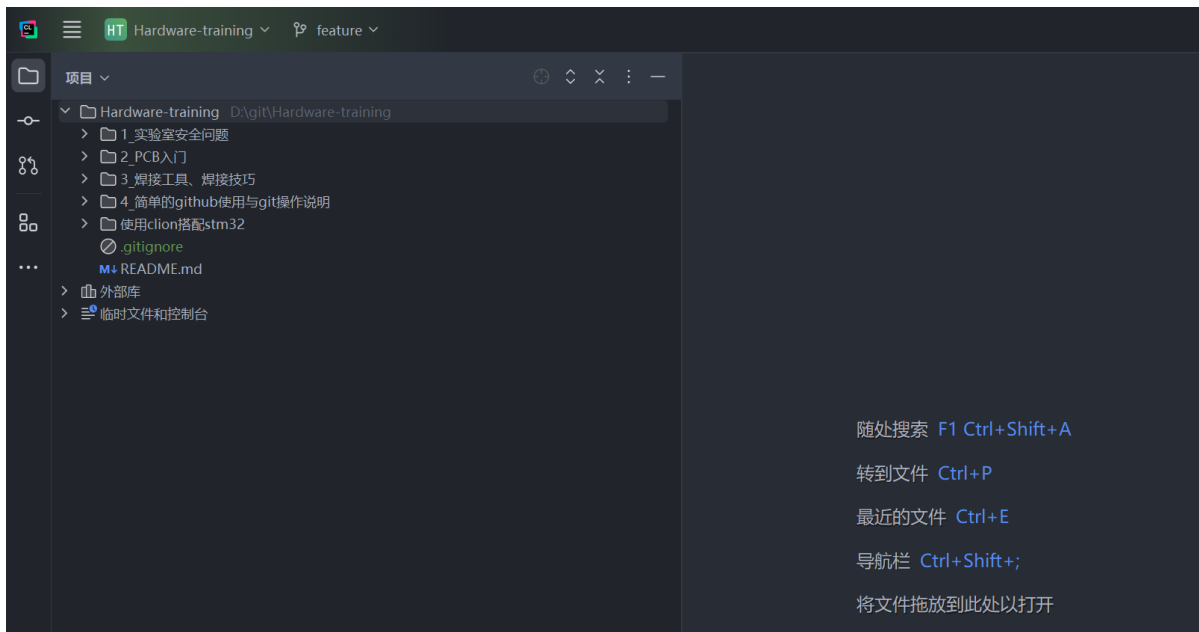


只要出现旁边那个已下载固件，就算成功，左边的虽然是红色字体，但不是报错

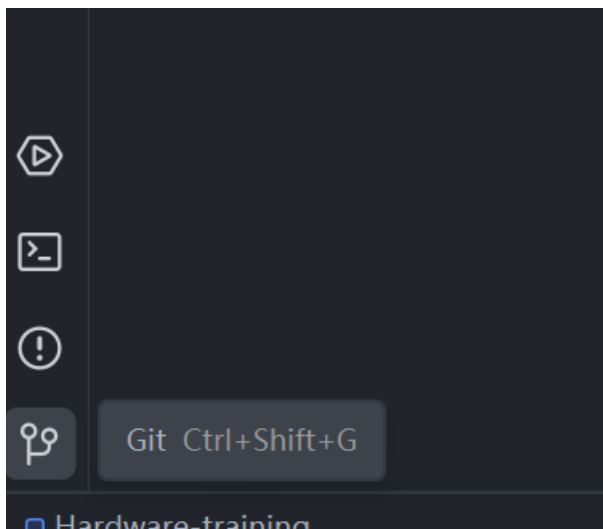
4 使用clion管理git

4.1 在clion中打开整个仓库

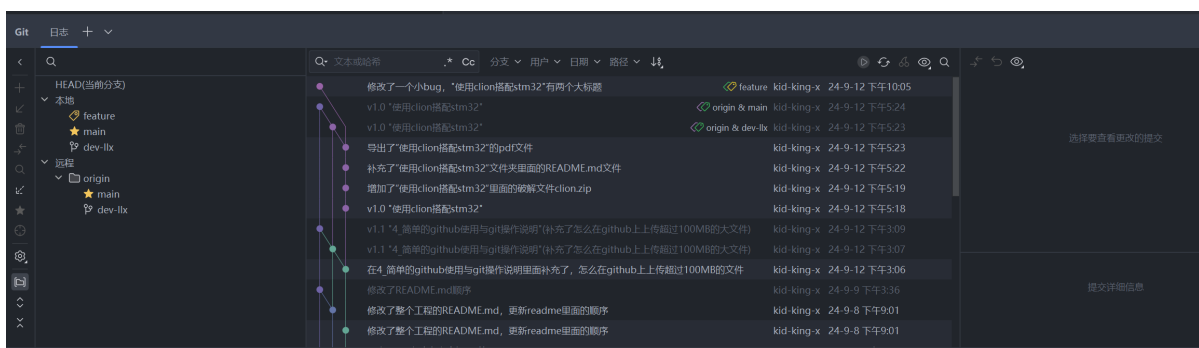
- 直接打开那个仓库的整个文件夹



- 然后可以看到那个仓库下面的所有文件，左下角有个Git分支类似的图标



- 点击那个图标可以看到，整个仓库的git分支图

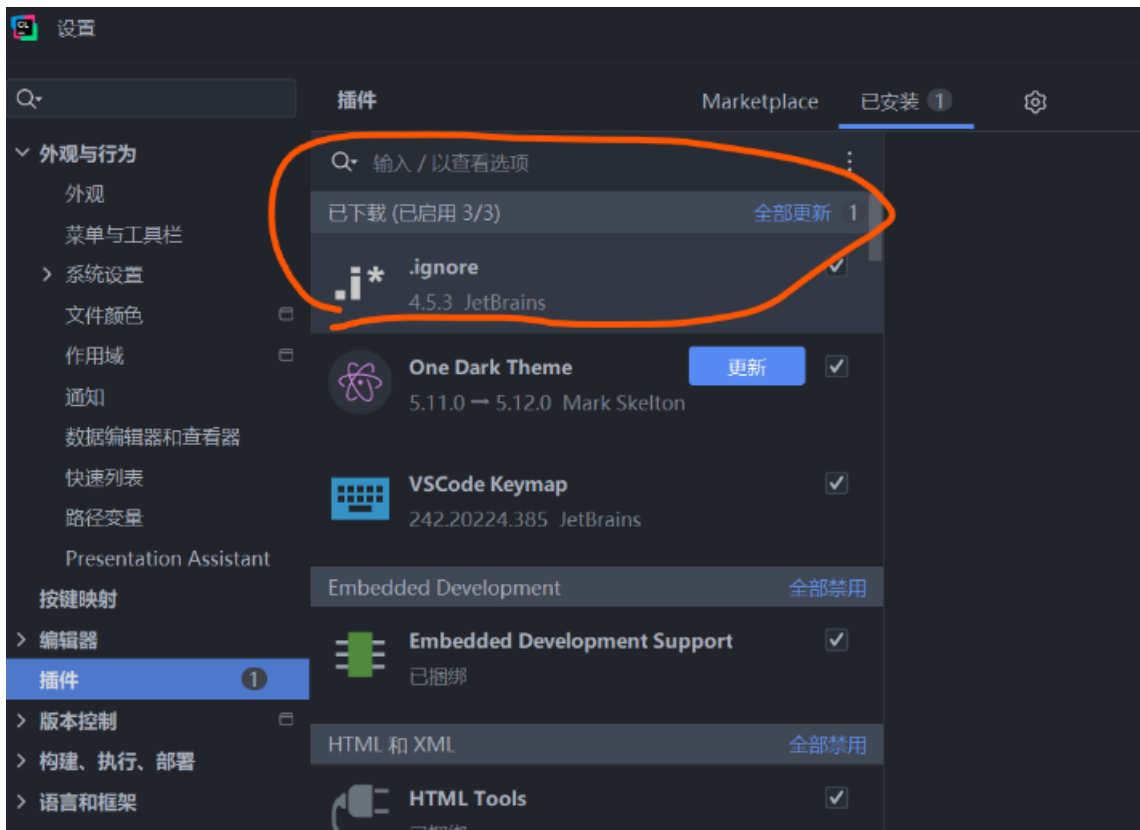


4.2 在clion中给仓库添加.gitignore文件

- 如果忘了.gitignore文件的，回到“简单的github使用与git操作说明”那节
- 先在插件这边下载.ignore
 1. 文件——>设置



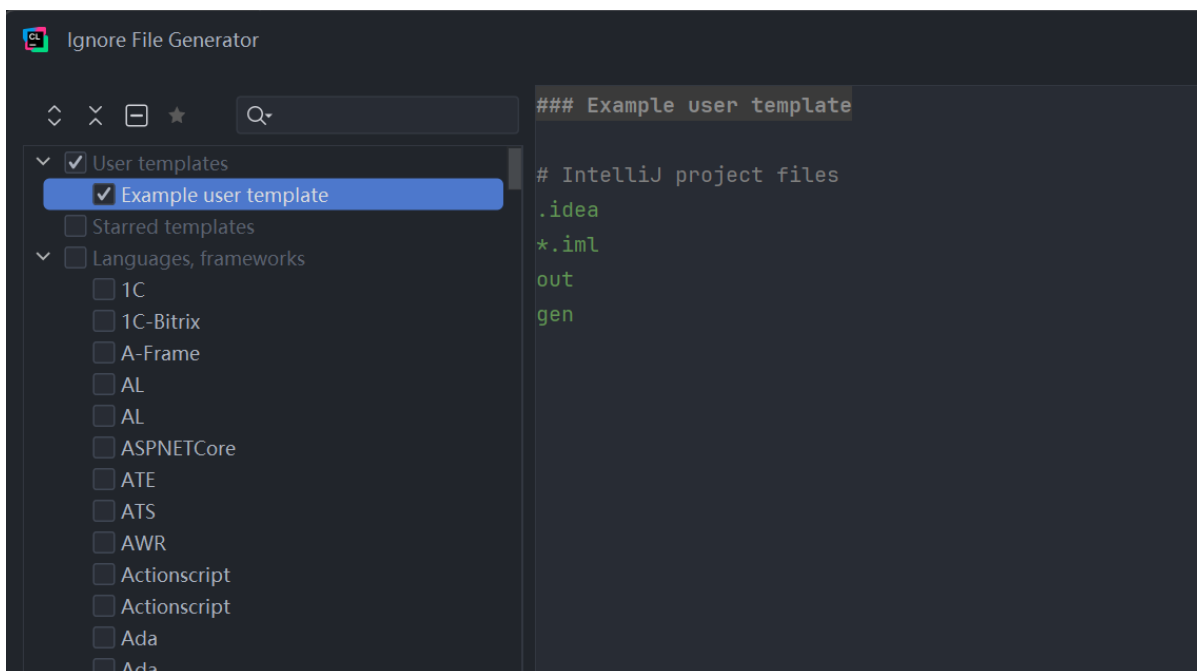
2. 点击插件，然后下载.ignore



- 下载完.ignore，叉掉设置，回到左边的项目，新建文件（右键点击——>.ignore File ——>.gitignore File(Git)）



- 点击Example user template, 再点击Generate

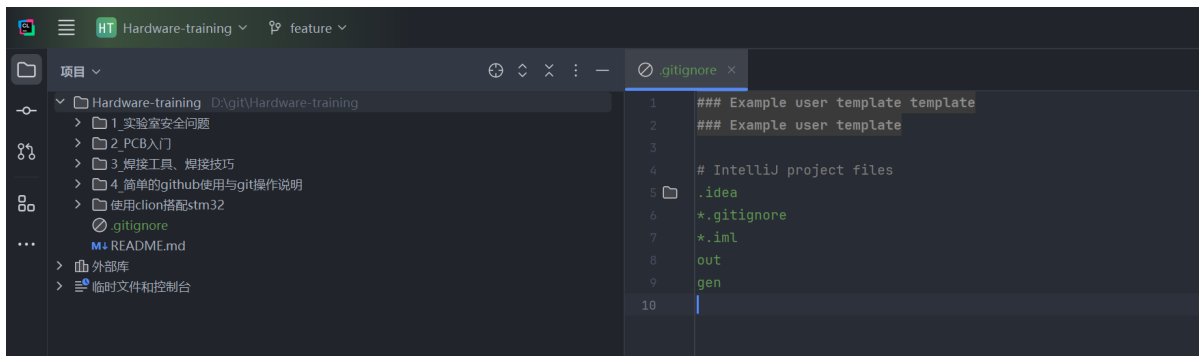


- 能看见生成了.gitignore文件，里面的代码如上图，对这里的代码进行修改，修改成下面这样

```

1  ### Example user template template
2  ### Example user template
3
4  # IntelliJ project files
5
6  # 整个.idea文件夹都不会上传
7  .idea
8
9  # 后缀为.gitignore文件不会上传，后缀为*.iml文件不会上传
10 *.gitignore
11 *.iml
12 out
13 gen

```



这样在git管理的时候，里面的文件/文件夹就不会上传了