ECE373 Lab1 Additional Contents

IDE Selection

If you are Mac or Linux users, please contact TAs.

Keil MDK 5

Irreplaceable choice for ARM debugging.

CLion

Strong C/C++ IDE， you can use either `ST-Link GDB Server` of `CubeCLT` or `OpenOCD` for downloading and debugging.

STM32CubeIDE

Just Eclipse, not recommended.

VSCode+CubeCLT

It takes some time to configure. Only when you are quite familiar with VSCode.

VSCode+OpenOCD

?

Keil Studio MDK 6

The Arm Debugger MDK 6 used seems to be worse than GDB.

Setup

IMPORTANT!! Make sure that:

- No Chinese characters should appear in the installation PATH
- Under any condition it is NOT suggested to change the default installation path.

Notices

You can come back to this section later after you complete the installation.

- You can use either Keil5 or CLion for editing and debugging one project. They will share the same folder but uses different tool chain to compile and debug.
- After you have generated code for once, do not use the option `STM32CubeIDE` because this will overwrite the `CMakeLists.txt` if you have modified it. Use `MDK-ARM` instead and you could use Keil5 to debug.
- If you are running an old version of STM32CubeMX, when generating code using the STM32CubeMX, do not choose `SW4STM32` for the `Toolchain / IDE`, otherwise CLion could not detect the project correctly.

STM32Cube

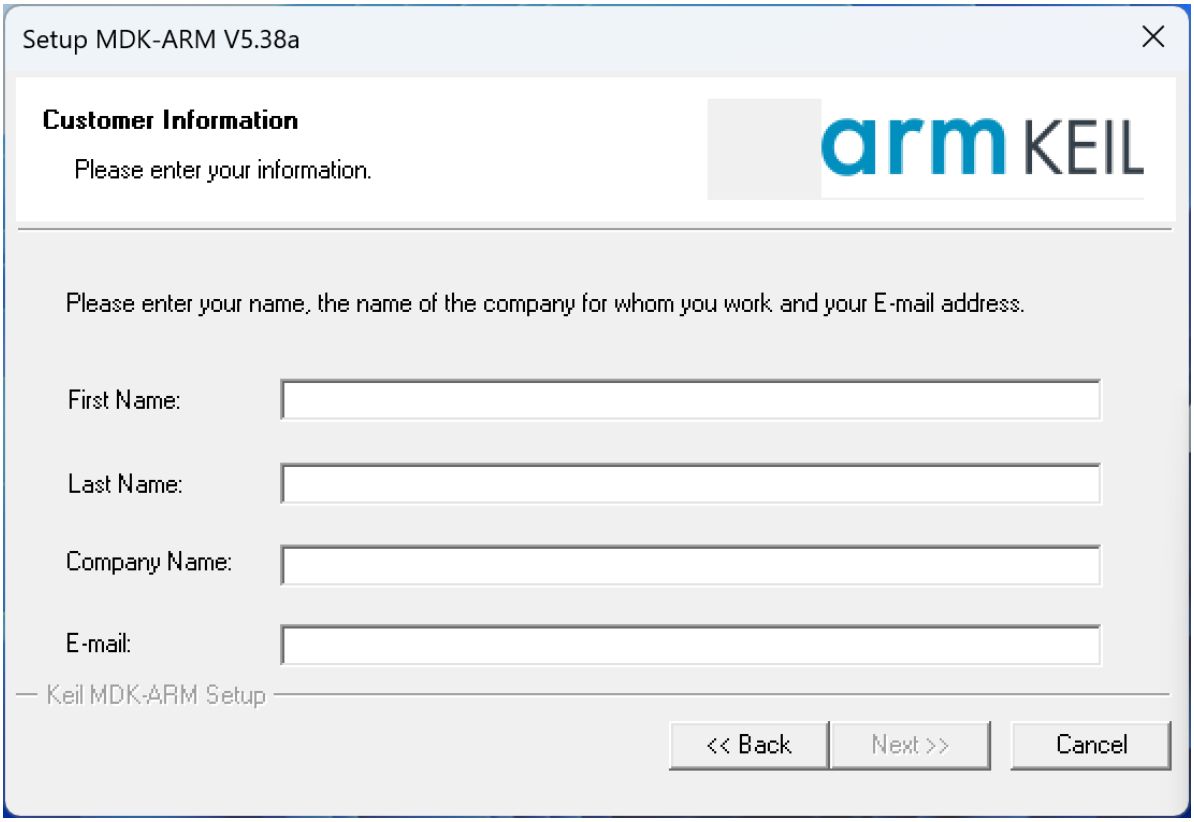
You may need an ST account to download and get access to the software.

Unzip then install directly, do not change installation path of `STM32CubeCLT`.

Keil MDK 5

Installation

Open `MDK538a.EXE`, enter whatever you want in these blanks.



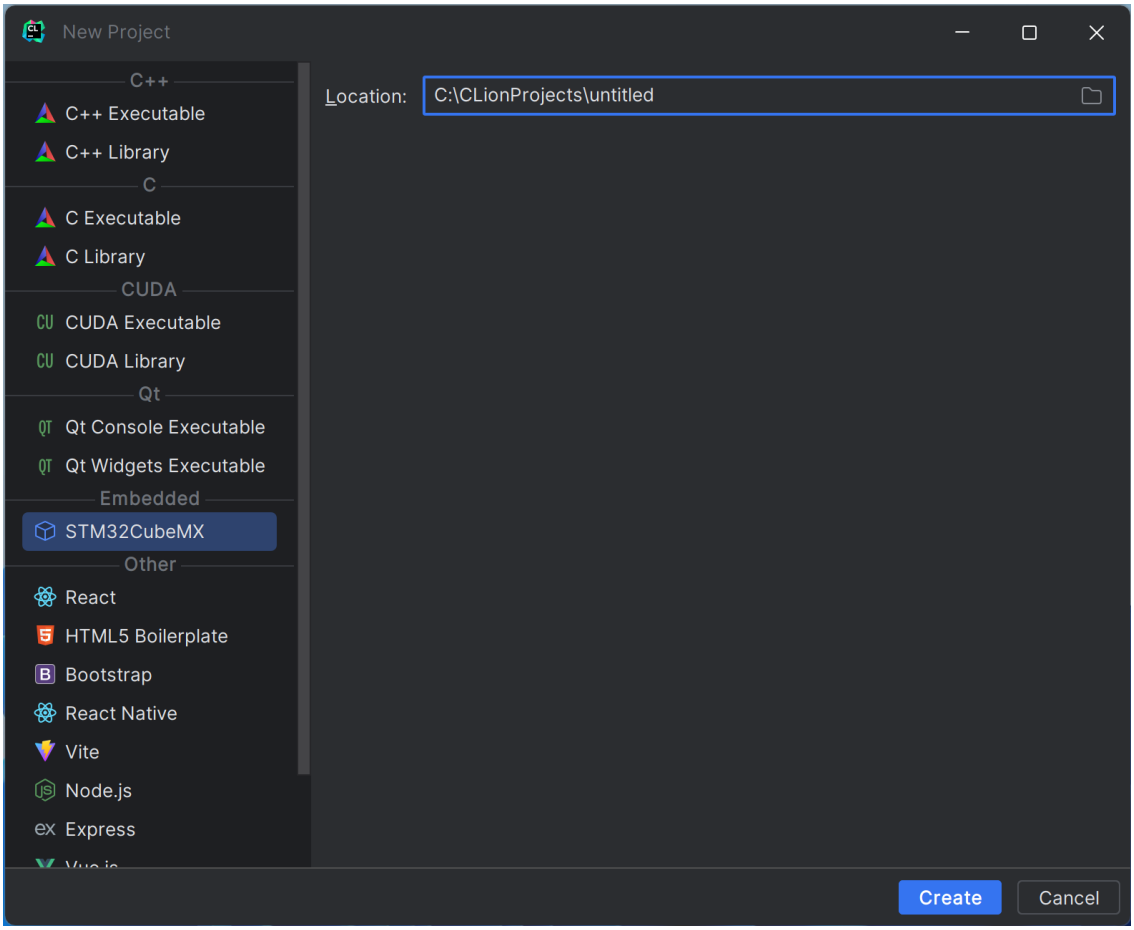
A Pack Installer window will pop up after your installation, just close it.

Activate

- Open Keil uVision5
- In `File` tab，find `License Management`
- Copy the `Computer ID` under `Single-User License`
- Open `keygen.exe`
- Select `ARM` as the `Target`，paste the `CID` just copied and `Generate`
- Enter the code to `New License ID Code (LIC):` in `License Management`，click `Add LIC`

CLion

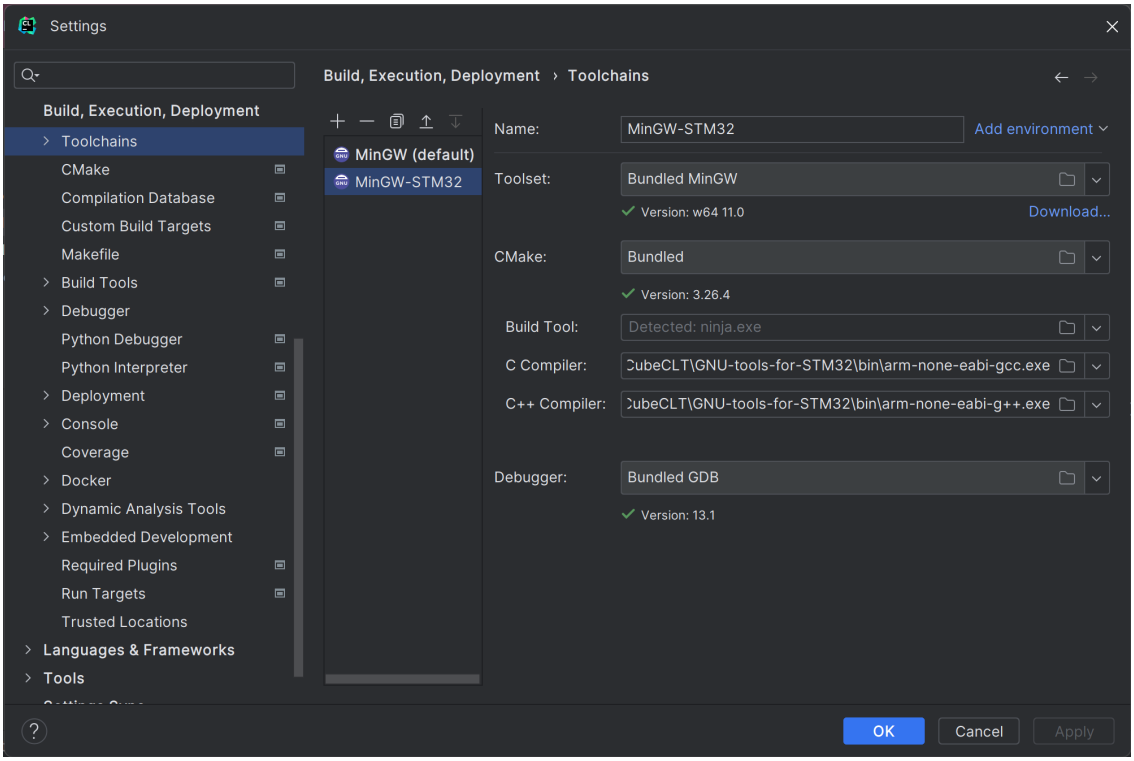
- Install CLion according to [上海交通大学软件授权中心 (sjtu.edu.cn)]
- Create a new `STM32CubeMX` project in CLion



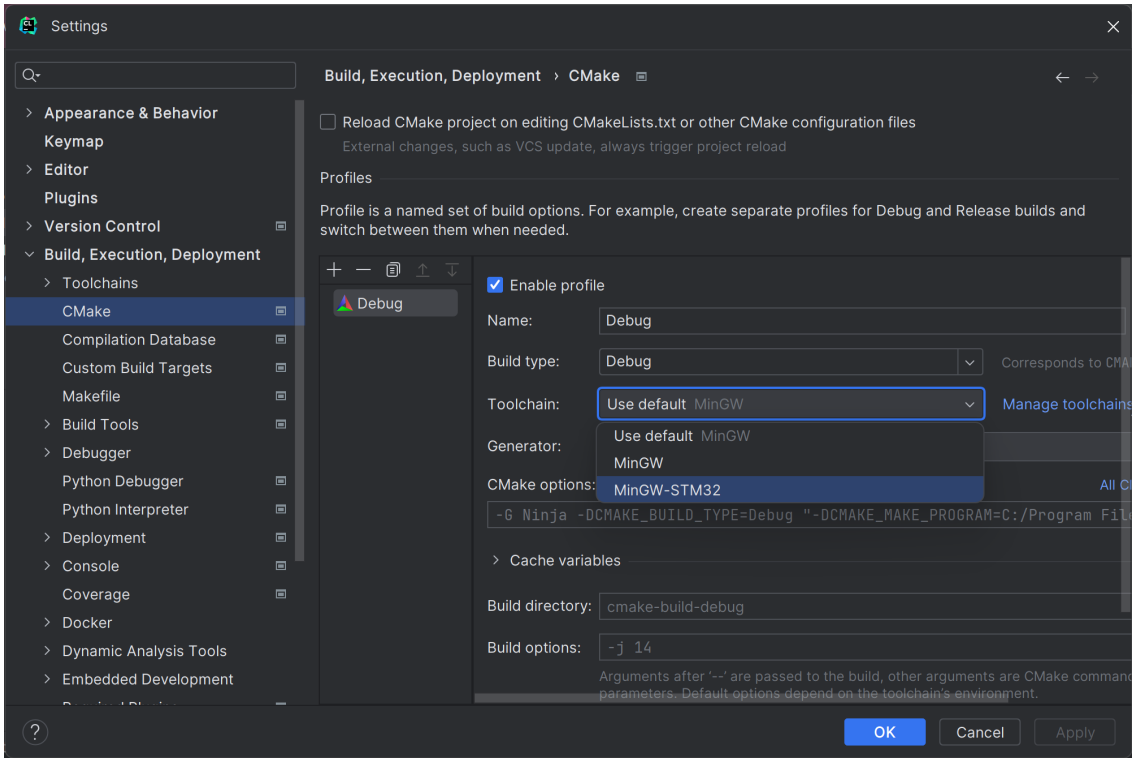
- Add a MinGW toolchain in `Toolchains` tab. Rename the toolchain and change the Compiler directory to:

  `C:\ST\STM32CubeCLT\GNU-tools-for-STM32\bin\arm-none-eabi-gcc.exe`

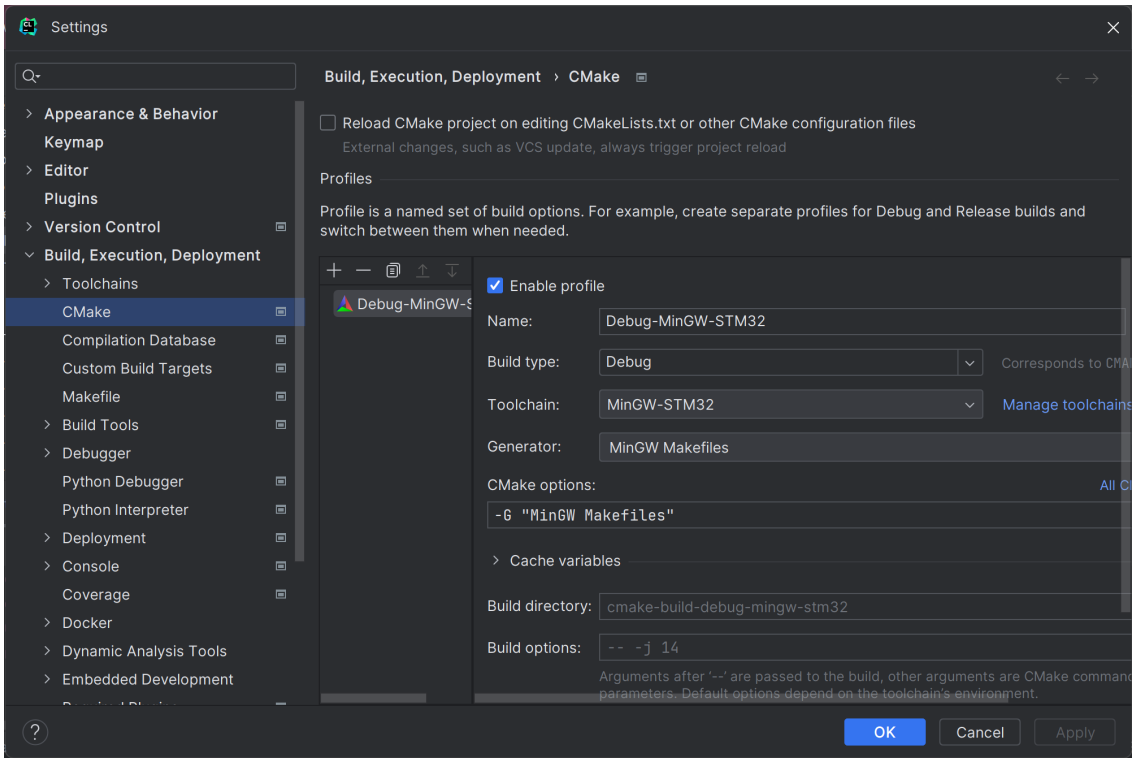
  `C:\ST\STM32CubeCLT\GNU-tools-for-STM32\bin\arm-none-eabi-g++.exe`



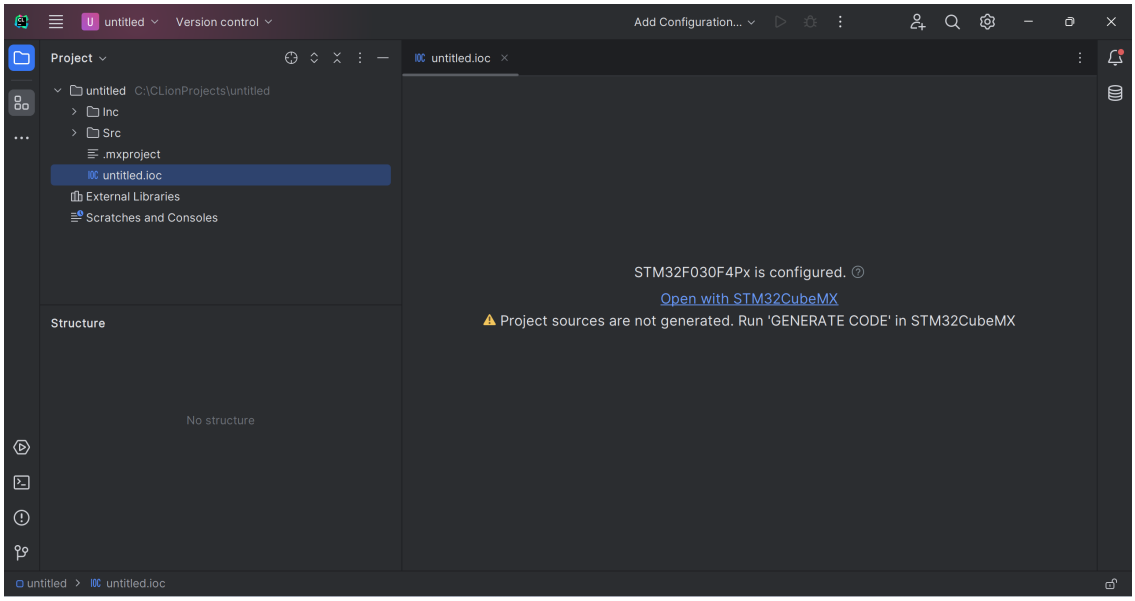
- For the project created, in `CMake` tab change the Toolchain to the one you've just created.



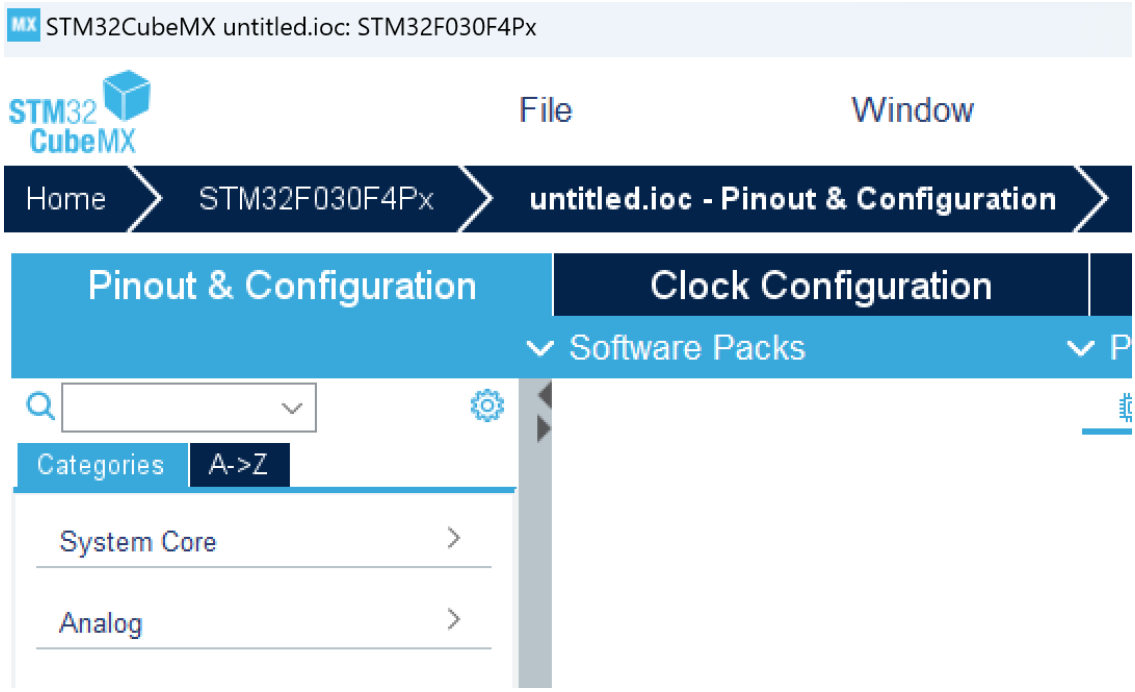
- Change the `Generator` to `MinGW Makefiles`



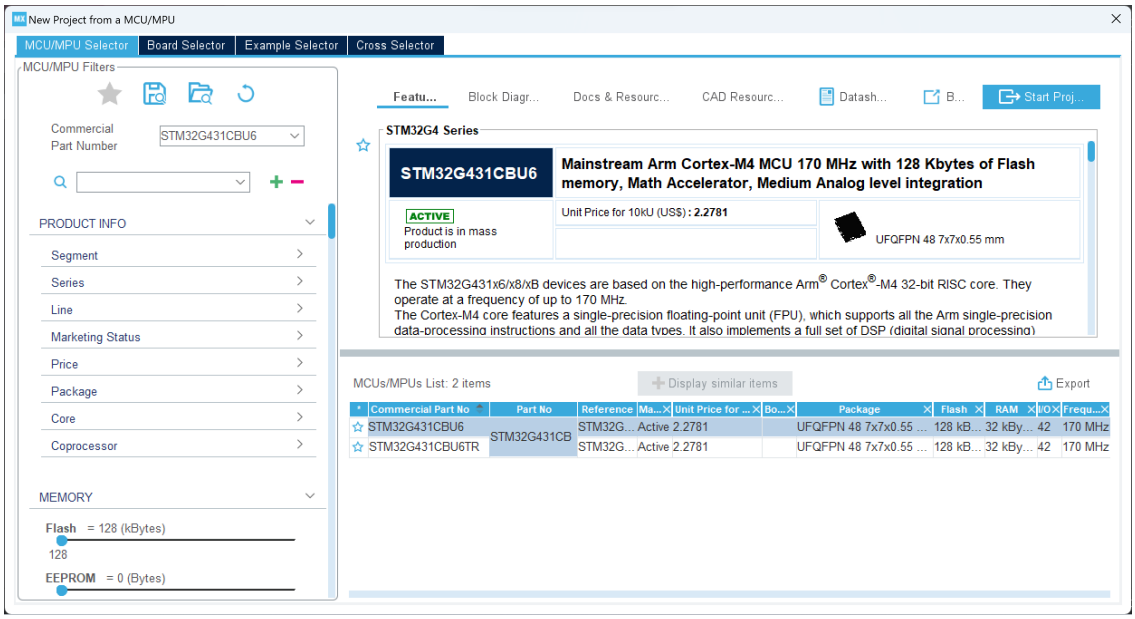
- After configuring the project you can find a `.ioc` file， click `Open with STM32CubeMX`



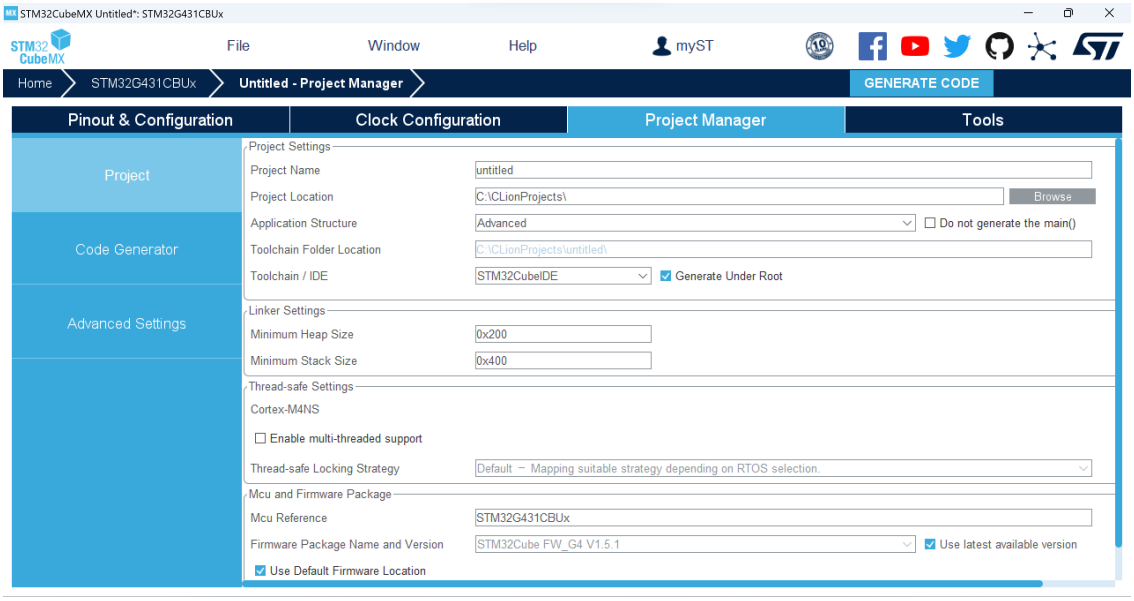
- After opening the STM32CubeMX, you can find that the default configuration is an F0 MCU, click the name `STM32F030F4Px` here to change the MCU



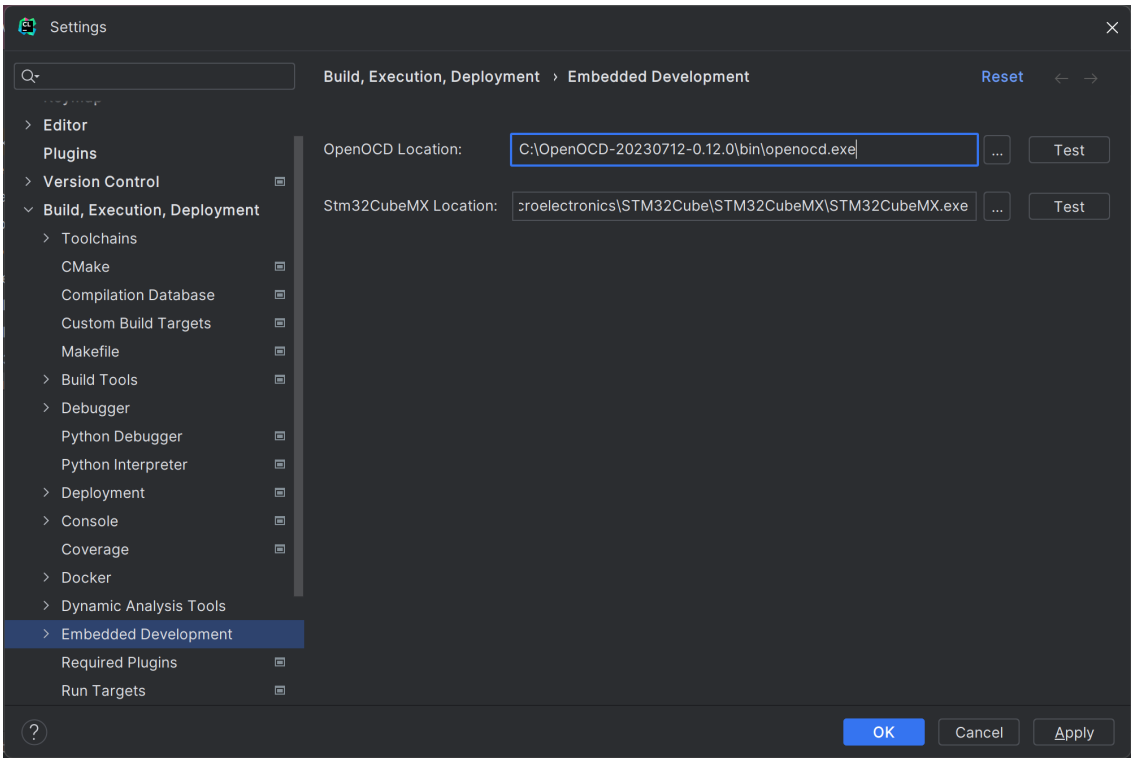
- Find and select the target MCU we want, here the example uses a `STM32G431CBU6`.



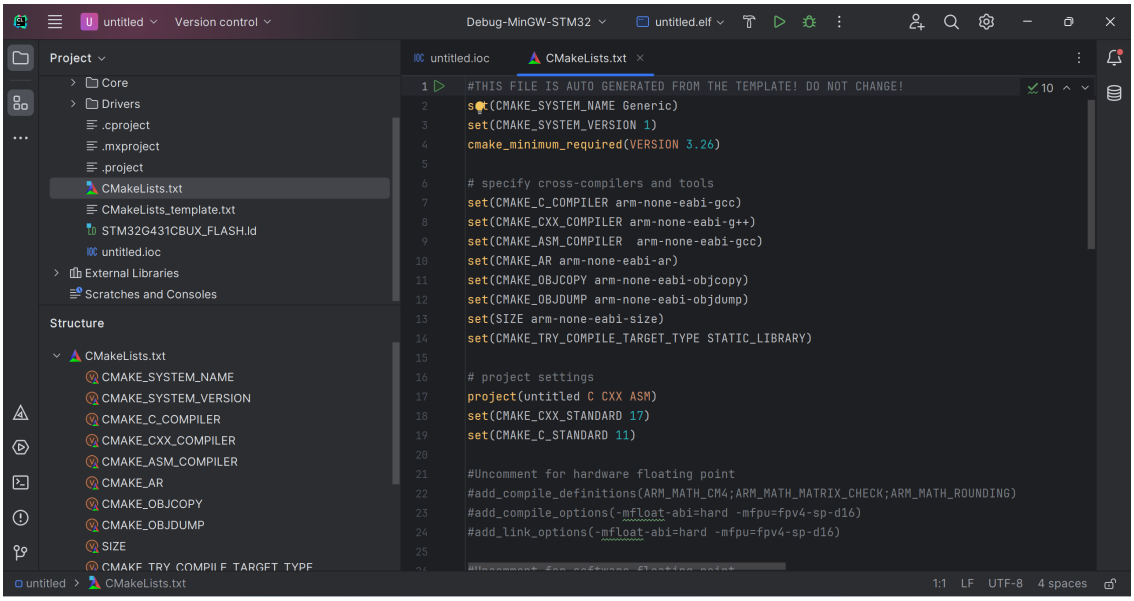
- In `Project Manager` tab, change the `Project Name` to the same name as the CLion project. Change the `Project Location` to the root directory of your CLion project (not your project directory). Choose `STM32CubeIDE` in `Toolchain / IDE` tab. After this click the `GENERATE CODE` and close the `CubeMX`.



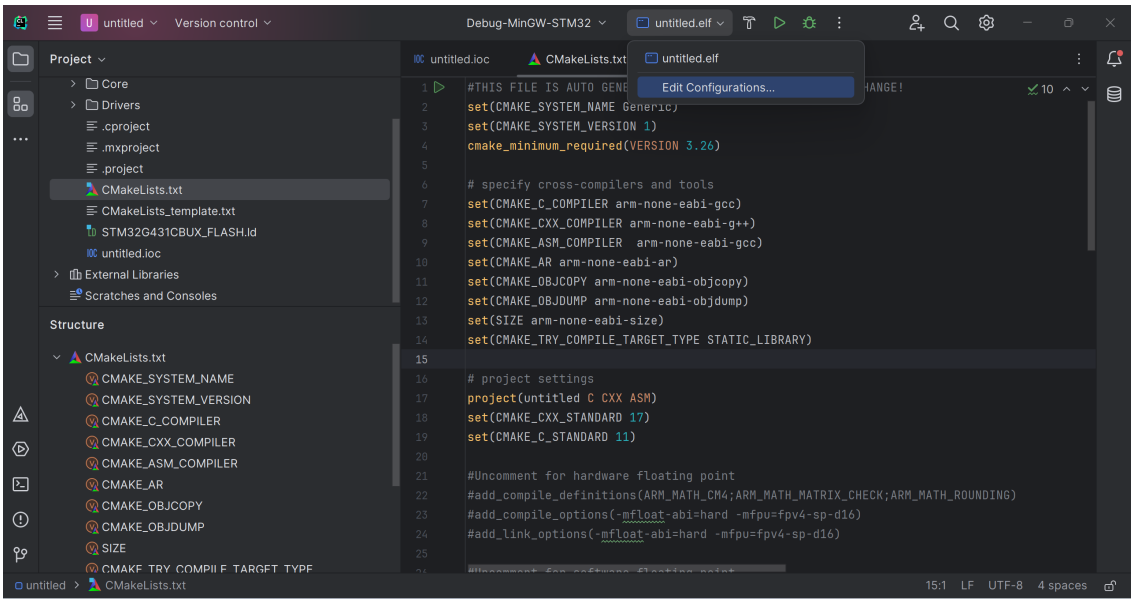
- Download OpenOCD from http://gnutoolchains.com/arm-eabi/openocd/. Configure the location of OpenOCD after unzip.



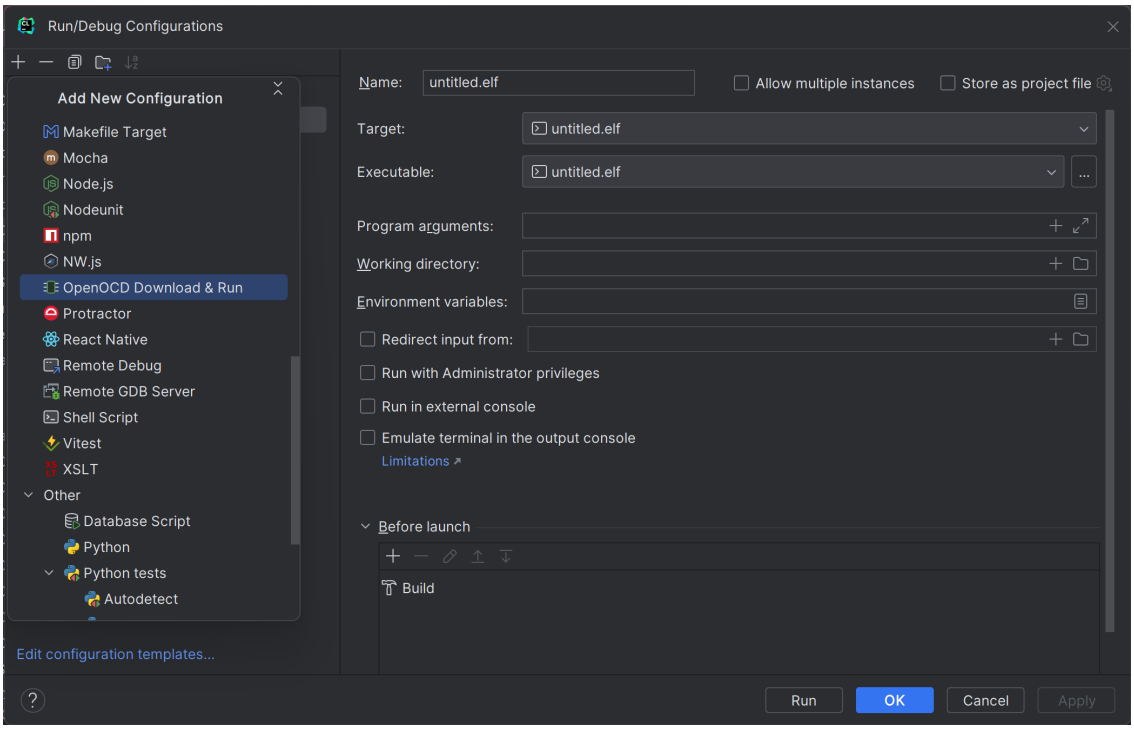
- A `CMakeLists.txt` will be generated.



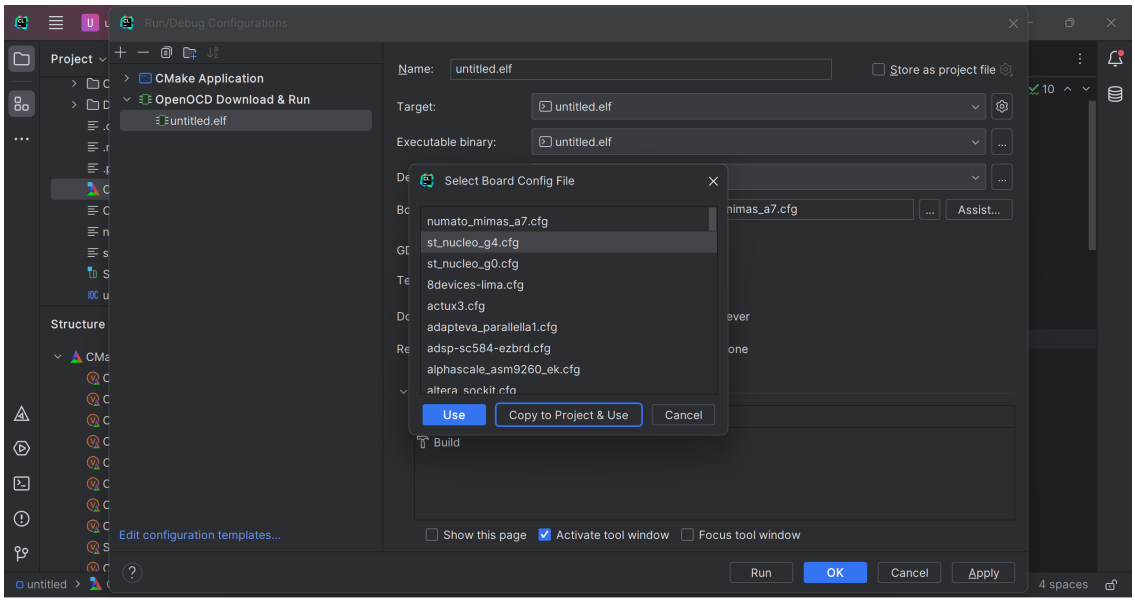
- Click `Edit Configurations` under the target selection tab.



- Add a `OpenOCD Download & Run` configuration, select the same `Executable binary` as `Target`



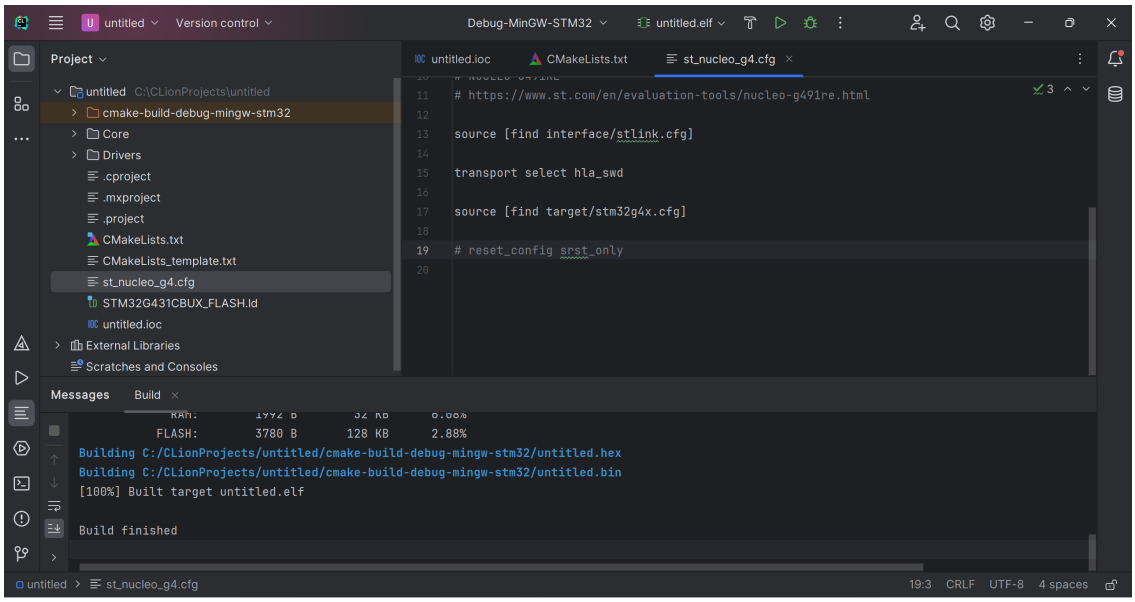
- Select a Board Config File
  For F1, you can use either `stm32f103c8_blue_pill.cfg` or `st_nucleo_f103rb.cfg`



- Comment the line

```
# reset_config srst_only
```

  Then build and run!



Additional Contents

If you could read Chinese, you can also refer to the following links.

[配置CLion用于STM32开发【优雅の嵌入式开发】 - 知乎 (zhihu.com)]

VSCode+CubeCLT

[【电赛-软件】基于ST官方插件 配置VScode开发STM32-CSDN博客]