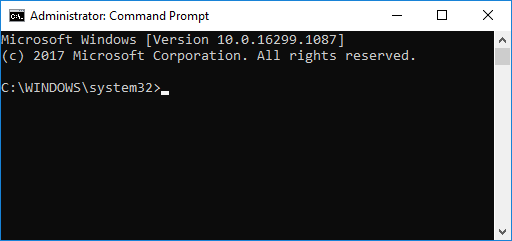
# 怎样创建zephyr工程？

软件要求：Win10，[MCUXpresso11.3.1](https://www.nxp.com/design/software/development-software/mcuxpresso-software-and-tools-/mcuxpresso-integrated-development-environment-ide:MCUXpresso-IDE?tab=Design_Tools_Tab#nogo)，[LPCScrypt](https://www.nxp.com/design/microcontrollers-developer-resources/lpcscrypt-v2-1-2:LPCSCRYPT?&tab=Design_Tools_Tab)，[jlink v7.22](https://www.segger.com/downloads/jlink/)，python3

开发板：frdm\_k64

1 安装[Chocolatey](https://chocolatey.org/)

2 以管理员身份打开cmd（不要使用powershell）



3 输入以下三条指令

>choco feature enable -n allowGlobalConfirmation

> choco install cmake --installargs 'ADD\_CMAKE\_TO\_PATH=System'

> choco install git python ninja gperf

4 继续输入

>pip3 install -U west

安装west

5 开始clone仓库，userprofile是用户目录，一般是c/user/name这个目录，name是用户名字

> cd %userprofile%

> west init --mr v2.6.0 zephyrproject

> cd zephyrproject

> west update

west update有的文件可能下不下来，需要多试几次。

6 导出Zephyr CMake软件包。允许CMake自动加载构建Zephyr应用程序所需的样板代码

>west zephyr-export

7 安装python依赖

>pip3 install -r zephyr/scripts/requirements.txt

8 重新打开cmd，不需要管理员权限，确保已经安装了mcuxpresso 11.3.1，jlink版本大于V7

9 在userprofile下，创建一个文件zephyrrc.cmd，

里面输入如下，根据自己工具安装位置设定

set ZEPHYR\_TOOLCHAIN\_VARIANT=gnuarmemb

set GNUARMEMB\_TOOLCHAIN\_PATH=C:\nxp\**MCUXpressoIDE\_11.3.1\_5262**\ide\tools

set PATH=%PATH%;C:\Program Files (x86)\SEGGER\JLink**\_V722**

10 设置build环境

> cd %userprofile%\zephyrproject\zephyr

> zephyr-env.cmd

zephyr-env.cmd会调用zephyrrc.cmd，所以要确保有这个文件

11 创建led应用

>west build -b frdm\_k64f -d build\blinky samples\basic\blinky

会生成用于烧写的elf文件在build\blinky目录下

12 west flash –d build\blinky 这样就可以烧写这个文件

但是我们实际上需要一个完整的工程来开发代码，这时候可以这样生成工程

13 进入cd %userprofile%\zephyrproject\zephyr，使用zephyr-env.cmd，之前做过了可以省略

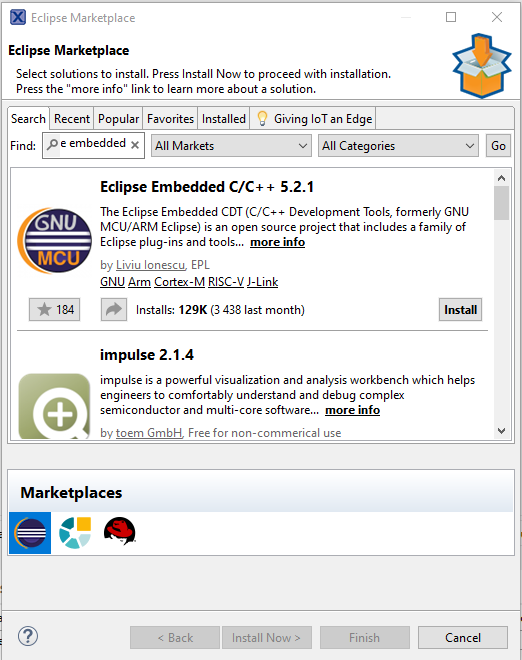
14 进入userprofile

>cd %userprofile%

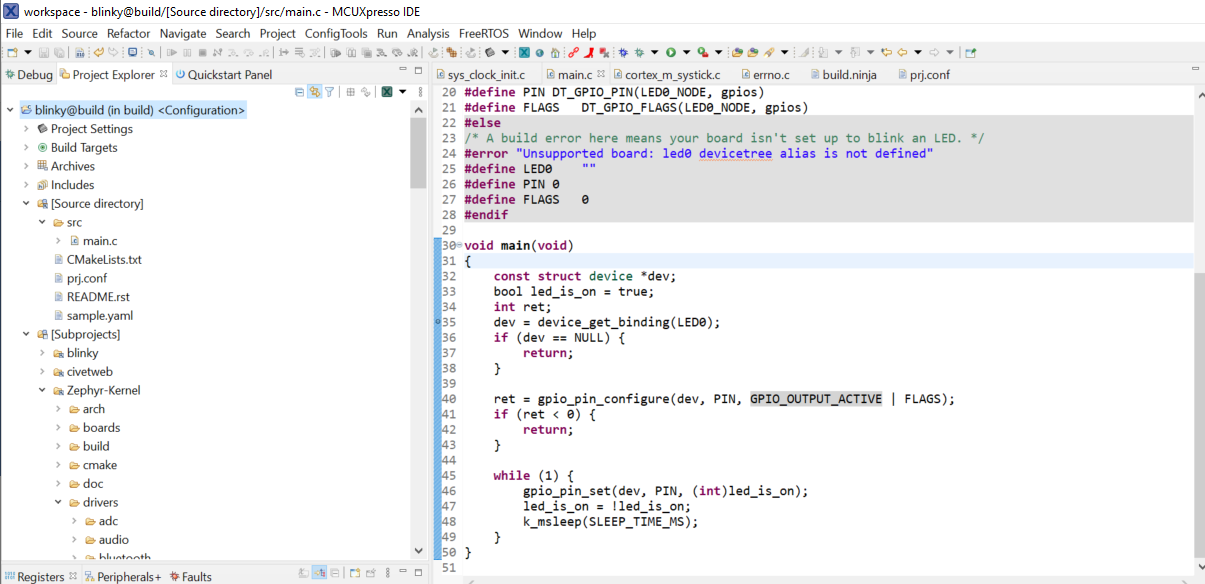
15 生成工程

> west build -b frdm\_k64f %ZEPHYR\_BASE%\samples\hello\_world -G"Eclipse CDT4 - Ninja"

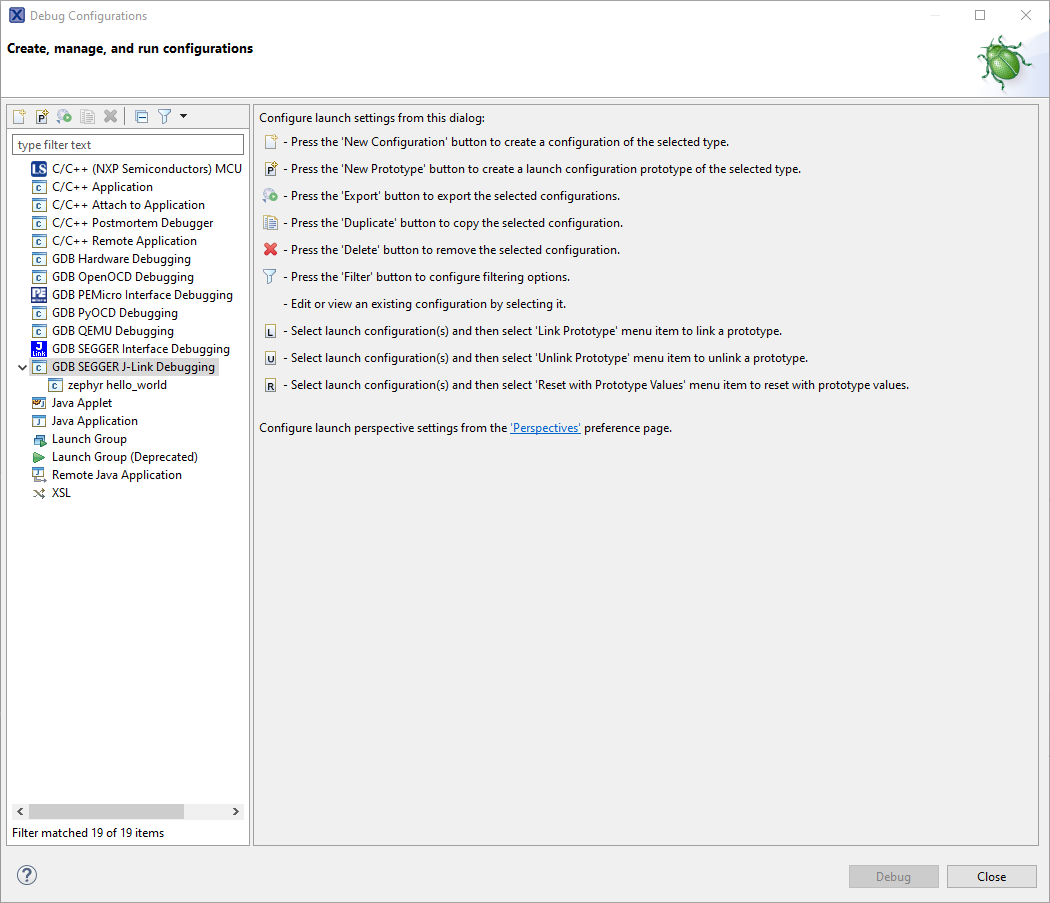
16 进入mcuxpresso，到help-> Eclipse Marketplace找到*eclipse embedded插件，安装*



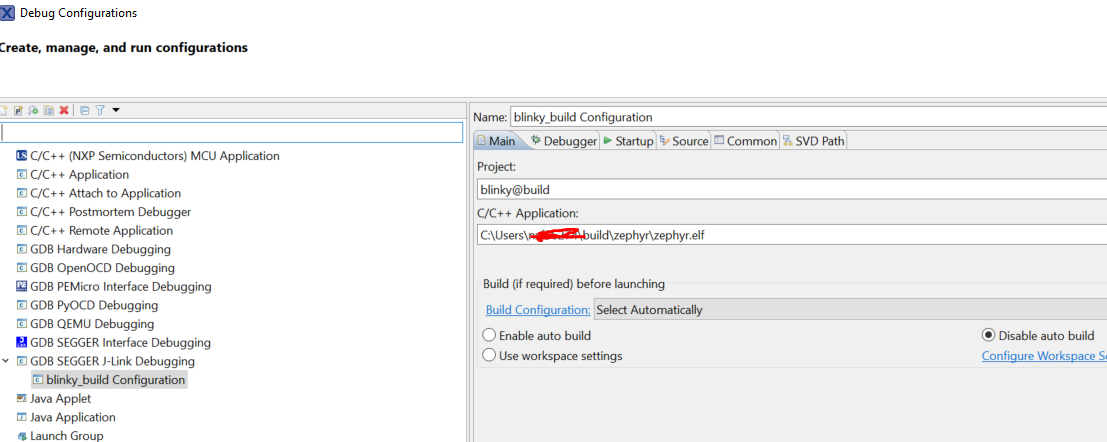
17 打开工程mcuxpresso，下面build和debug的按钮都是灰色的，无法使用，要build工程，要右键工程->build project



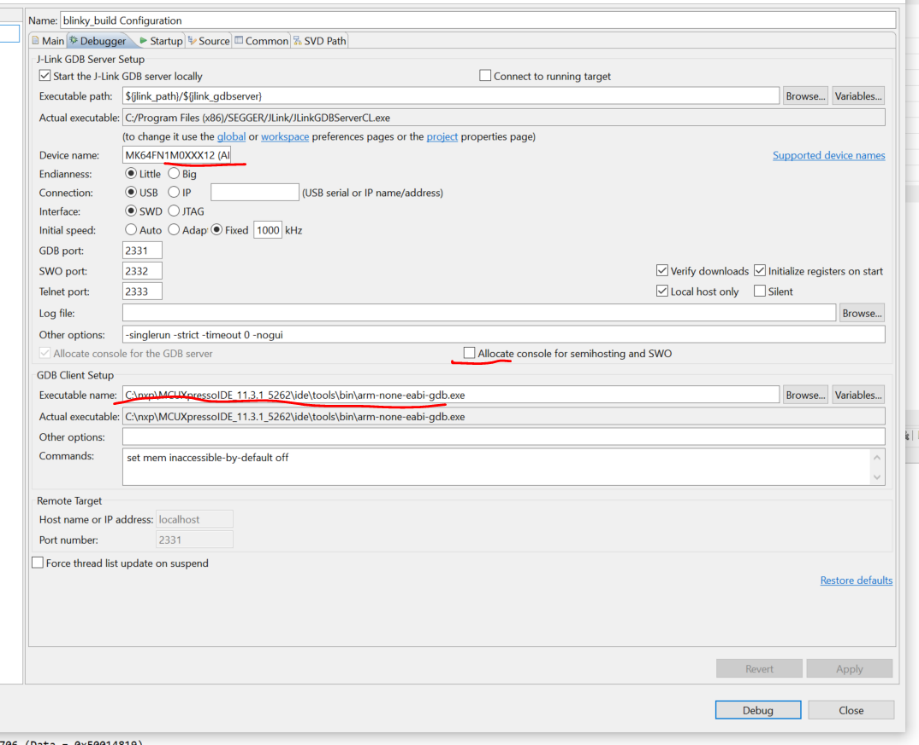
18 打开debug configuration（run-> Debug Configurations），新建一个标签在GDB SEGGER J-Link Debugging



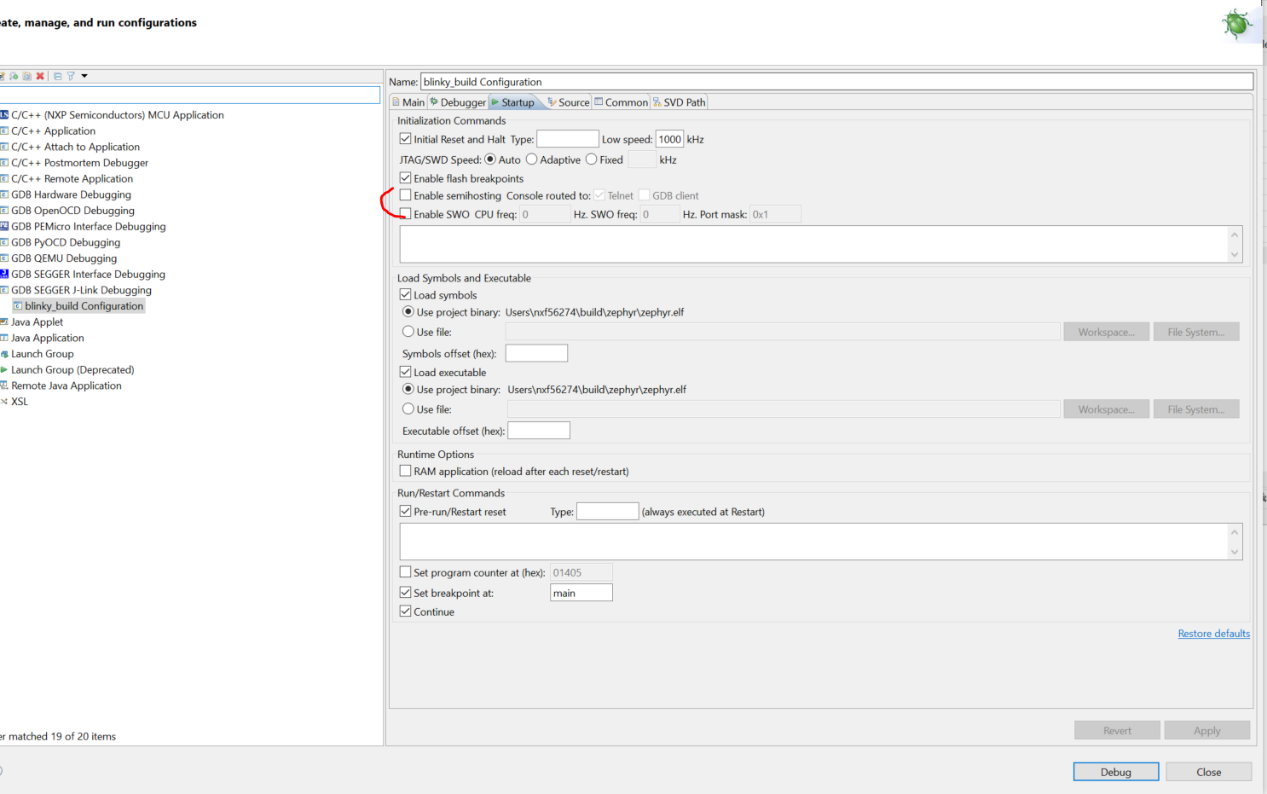
19 新建以后，在main标签里，将Project 和 C/C++ Application填好



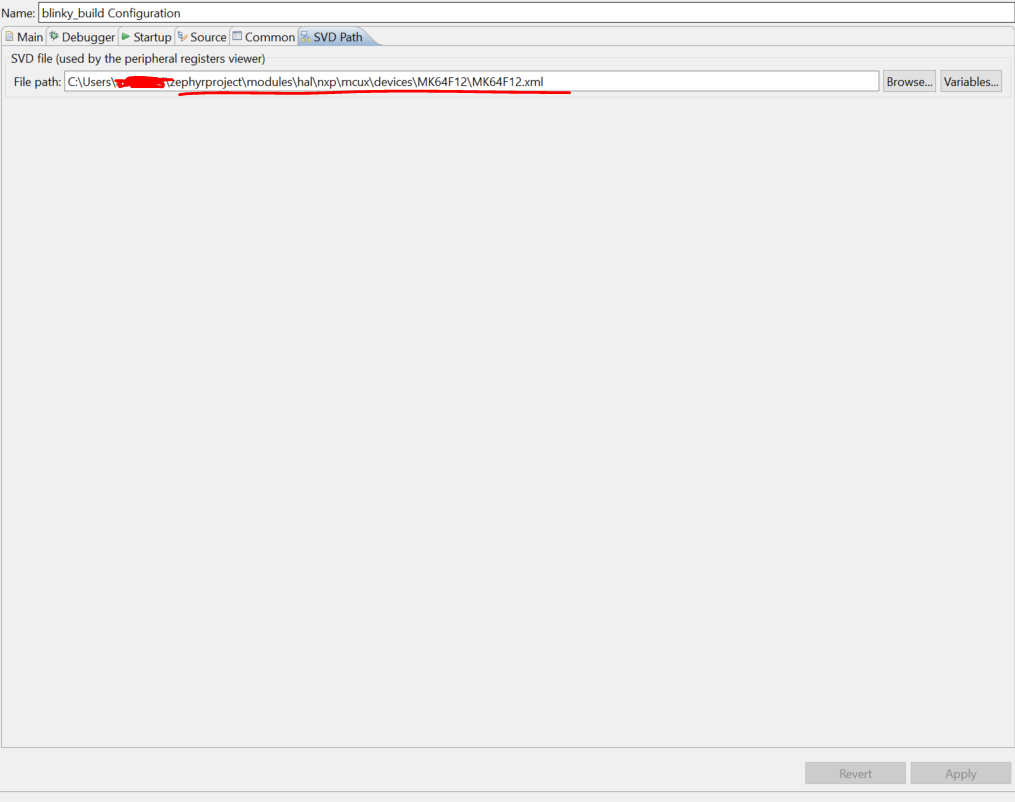
20 在debugger选项里，加入device name，以及其他



21在startup里，取消勾选



22 添加SVD文件，文件在zephyrproject\modules\hal\nxp\mcux\devices目录下



23 最后点击debug

