

提交版本	提交日期	提交人	说明
V0.1	2022 年 2 月 24 日	韦伟	初始验证版本
V0.2	2022 年 2 月 25 日	刘碧倩 王敬媛 罗英杰 喻刚桥	细化操作流程
V1.0	2022 年 2 月 26 日	黄超	提交版本
V2.0	2022 年 3 月 20 日	黄超	更换 voc 模型为 voc_roadsign 模型 (8bit)
V2.1	2022 年 5 月 10 日	黄超	修正推理时间为 880ms
V2.2	2023 年 3 月 24 日	王敬媛	更新细化操作流程

本文使用的环境为：

操作系统： Windows10

AIEP 远程调试工具： v4.0b

本文使用的环境为：

操作系统： Windows10

Quartus： 18.1 Standard

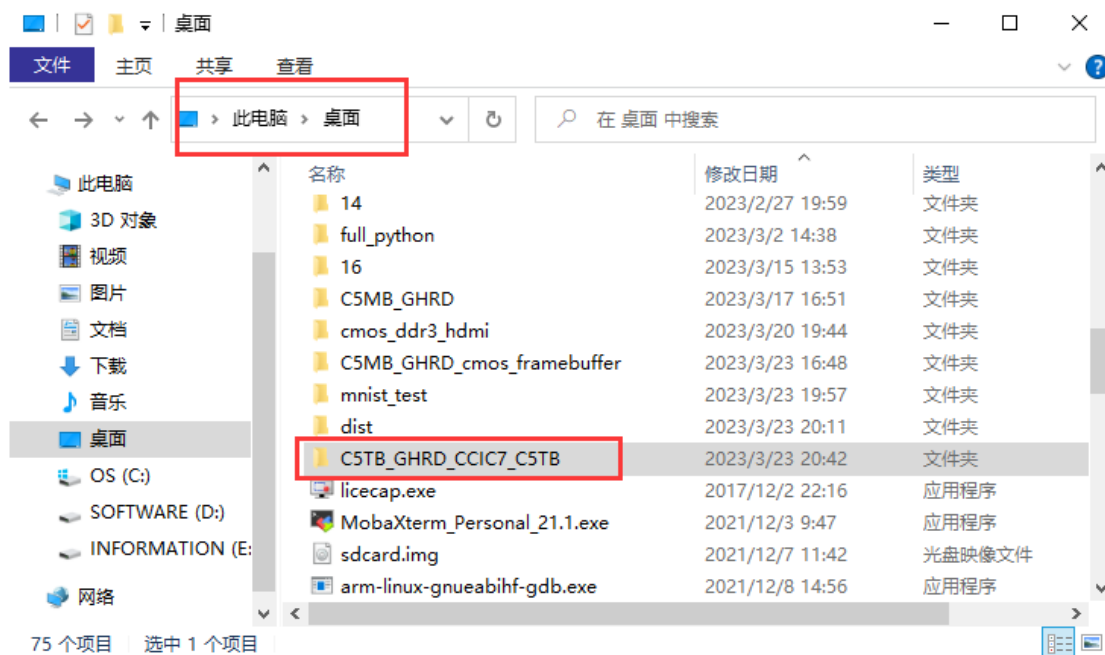
每个参赛团队均会分配到一个 VPN 账号，以及一个 AIEP 远程调试平台的 IP

以下正文

解压工程文件

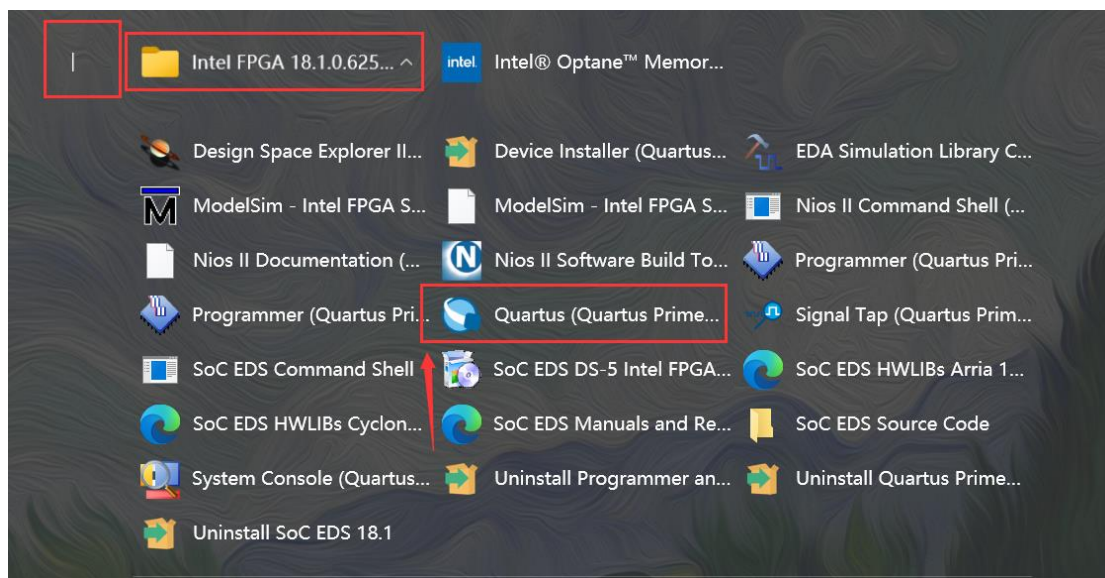
将提供的 PL 工程压缩文件 C5MB_GHRD150hls.zip 解压到计算机，注意路径中不要包含中文

和空格

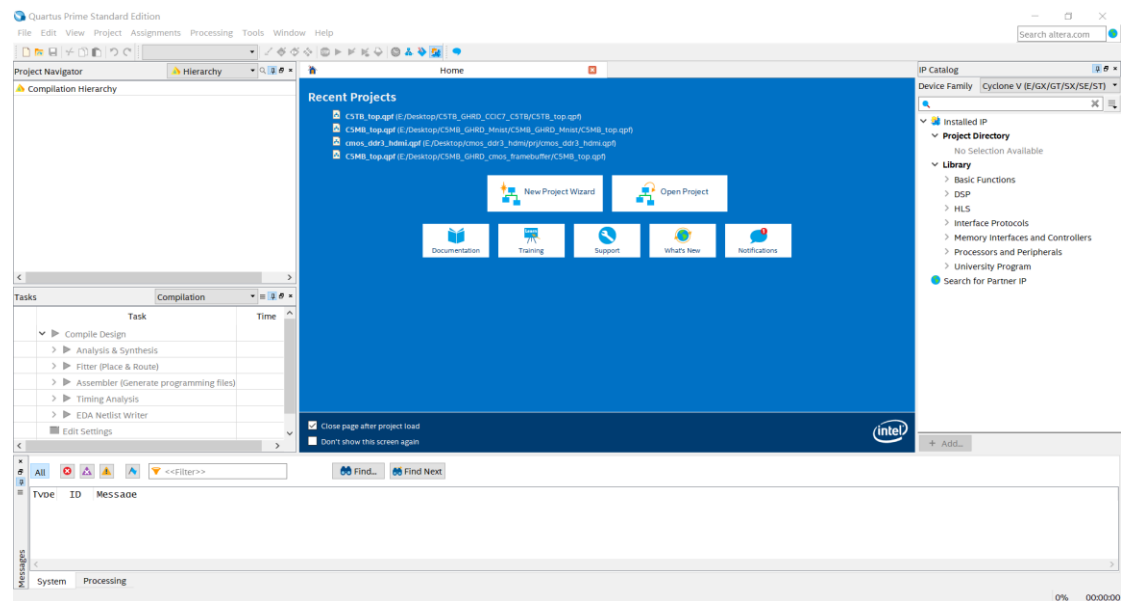


打开 Quartus

1、通过开始菜单打开 Quartus

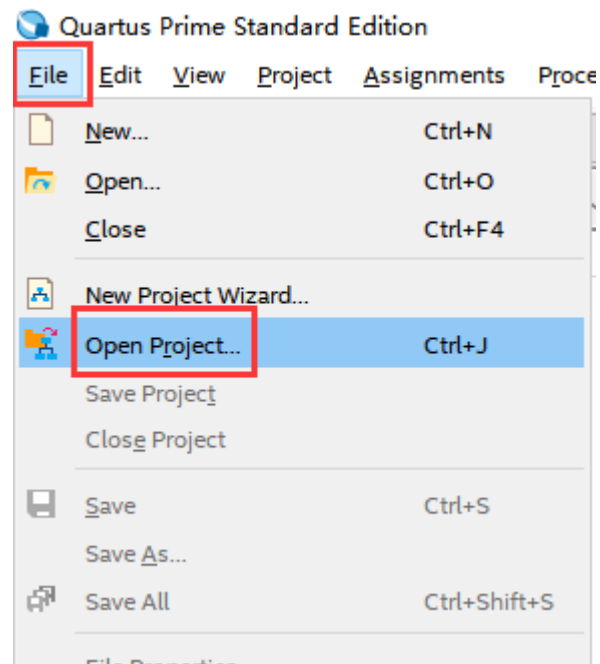


2、打开成功

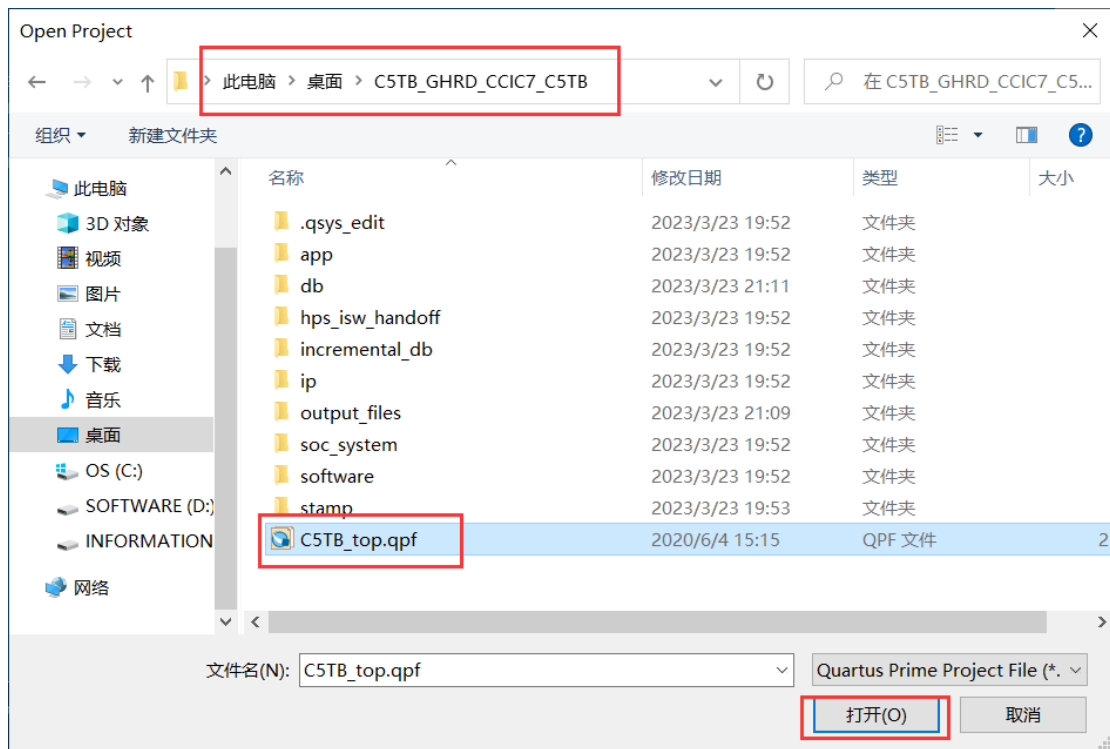


打开工程

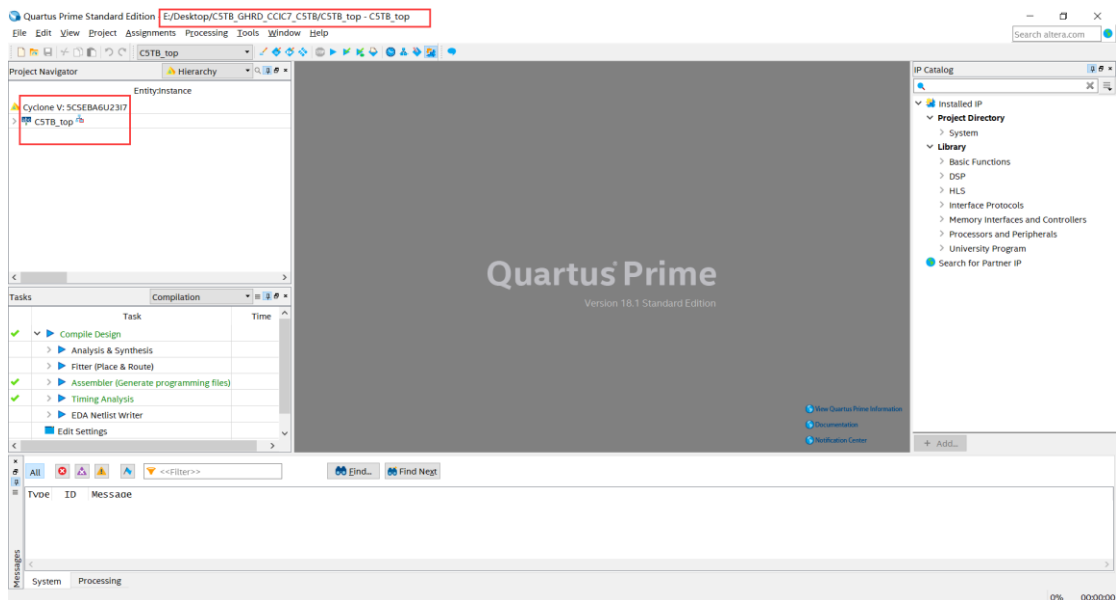
1、选择工程文件，打开工程



2、选择 C5MB_top.qpf 文件（之前解压的路径）



3、打开工程完成

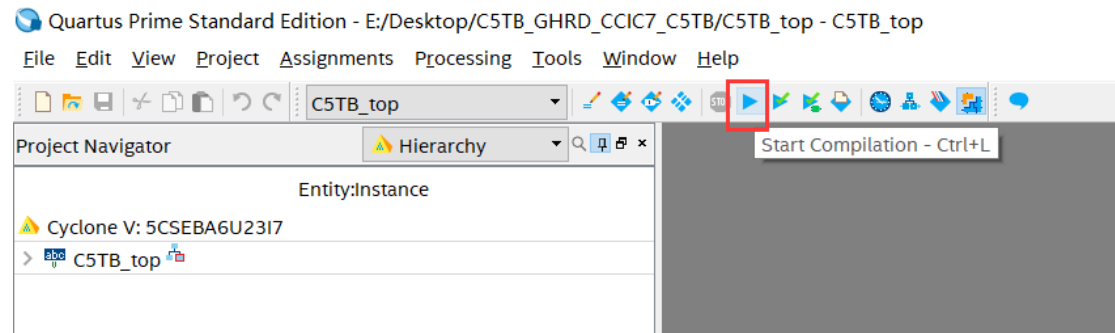


PL 工程界面

编译工程

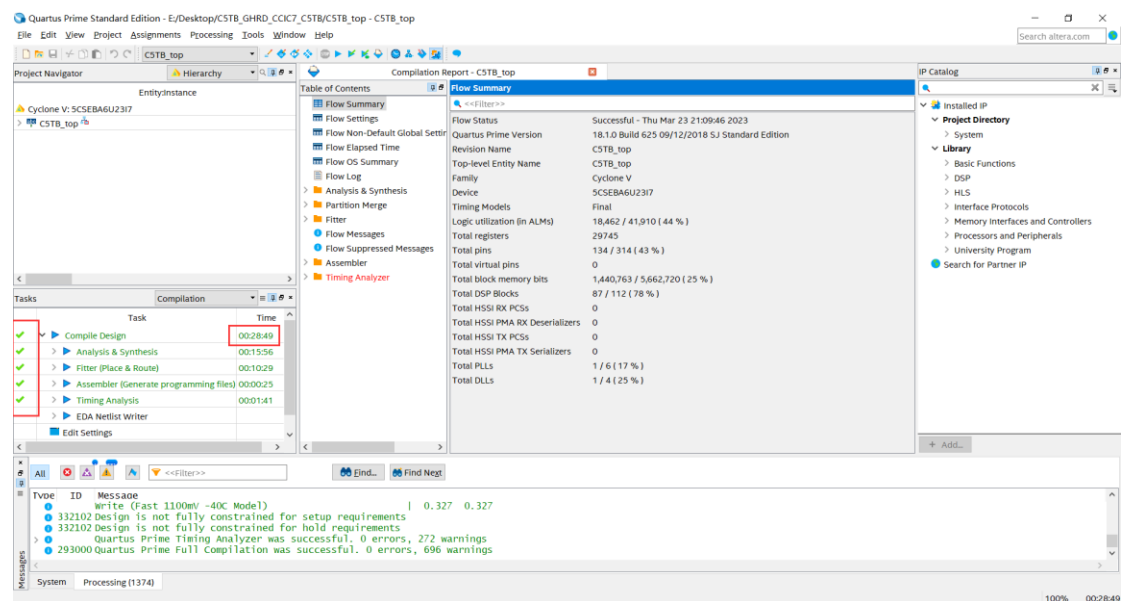
1、对 PL 工程进行全编译

点击“Start Compilation - Ctrl+L”按钮进行全编译



编译过程较长，请耐心等待

2、编译完成

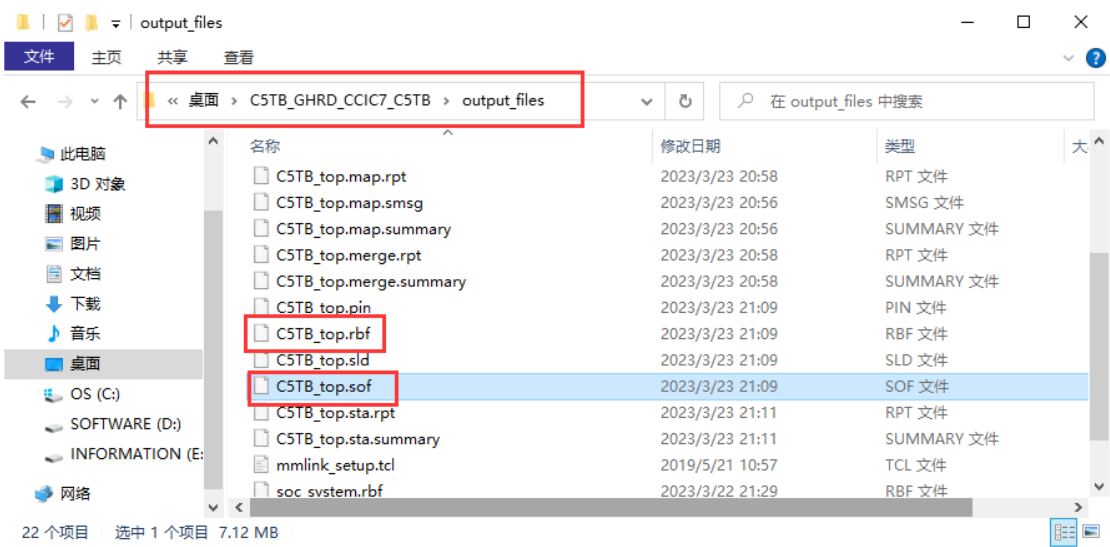


在我电脑上编译耗时 28：49

查看生成文件

经过以上步骤以后，我们可以在工程目录的 output_files 文件夹中找到生成的 sof 文件与 rbf

文件



注意两个文件的名字都为 C5TB_top，得到的 sof 和 rbf 文件就是我们需要用到的 FPGA 配置

文件

登录开发板

连接 VPN

VPN 连接方法参见 “重庆 OPEN VPN 使用手册 1.1.docx”

VPN 连接成功后如图，确保 VPN 如下图连接成功后再进行后续步骤



登录 AIEP

使用 AIEP 远程调试工具登陆 AIEP 开发板，详见《人工智能边缘实验平台-远程调试工具使用说明书(220105-gao)》

输入自己的账号密码登录

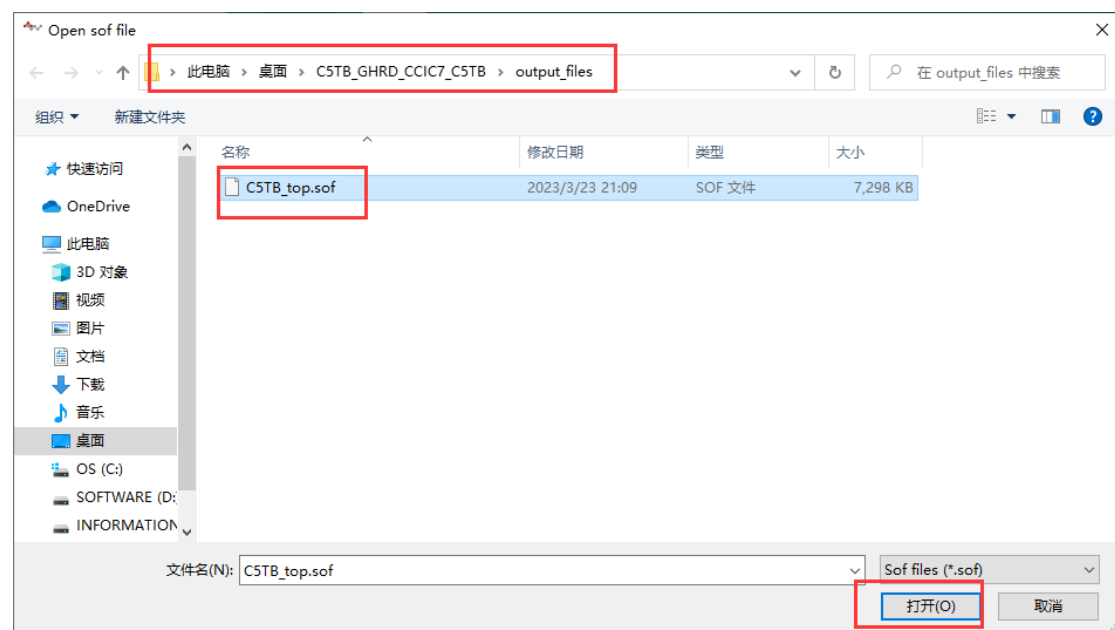
上传 Demo

上传 rbf 和 sof

点击 “选择 Sof 文件”



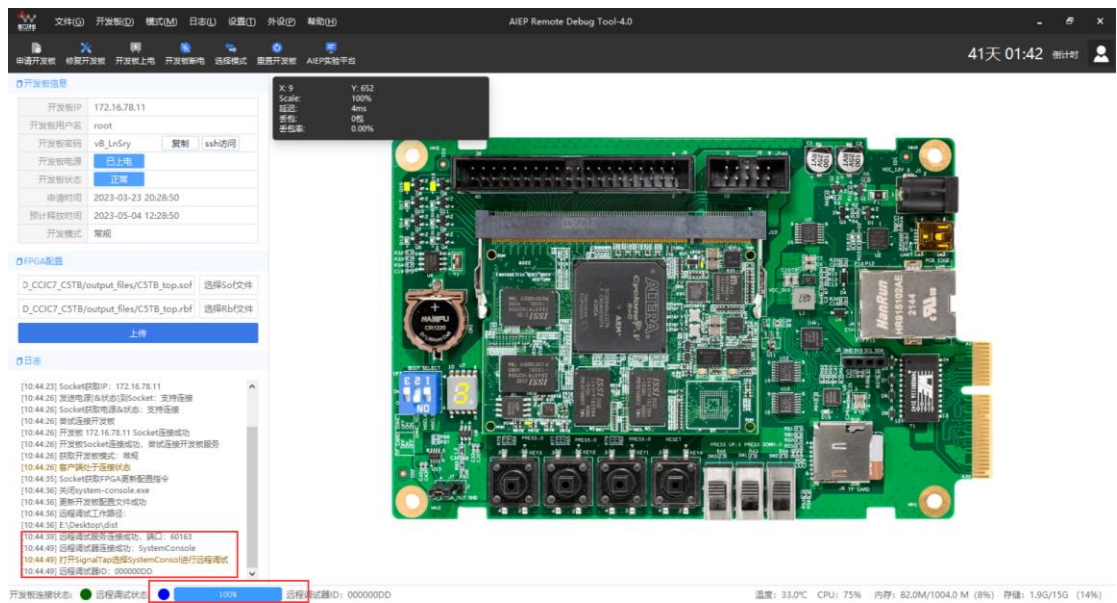
选择 sof 文件



点击 “选择 Rbf 文件”



上传成功

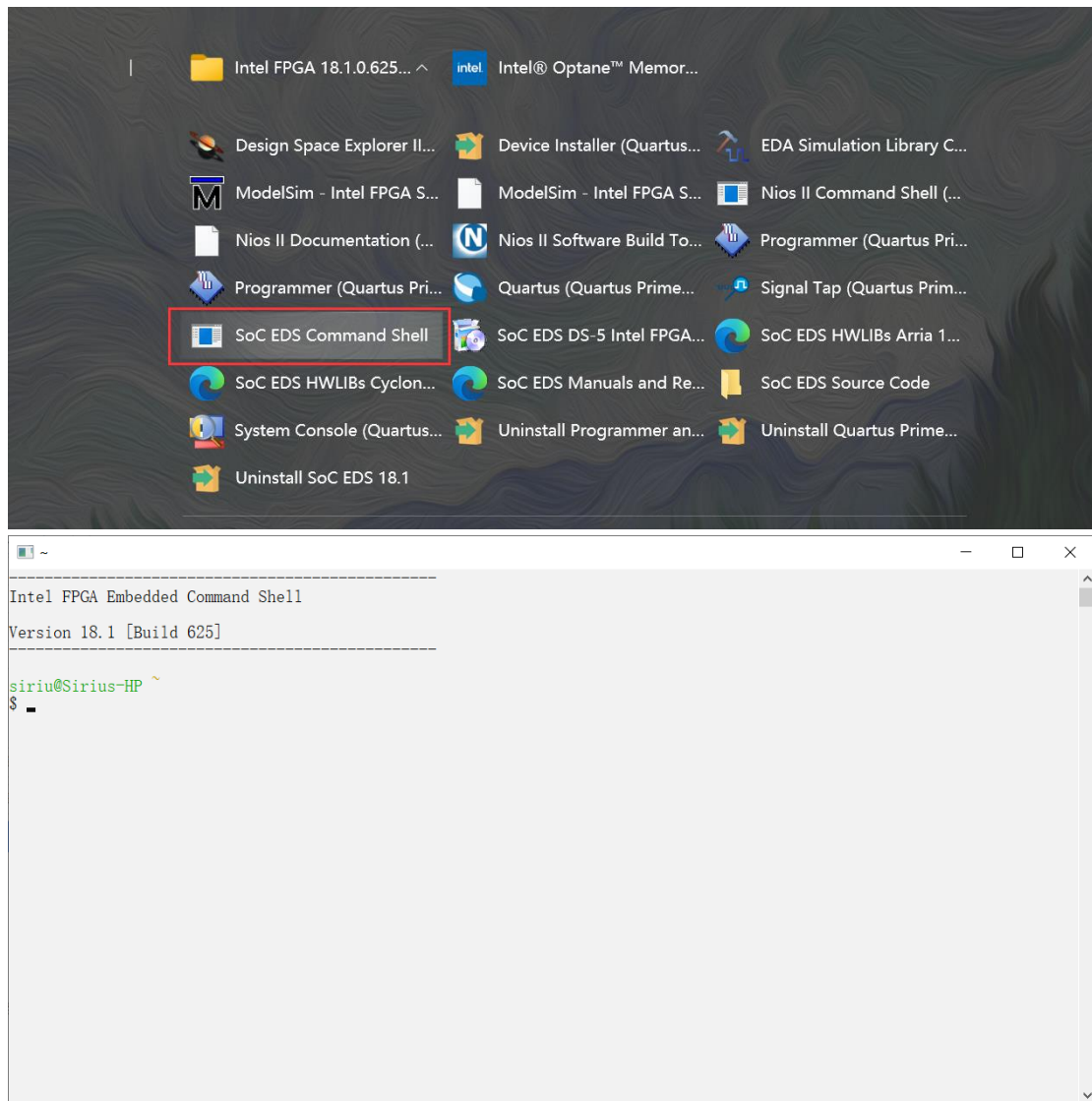


注意：上传 sof 及 rbf 文件后，请勿重启 AIEP，否则需要重新加载 sof 与 rbf

示例工程

打开 SoC EDS Command Shell

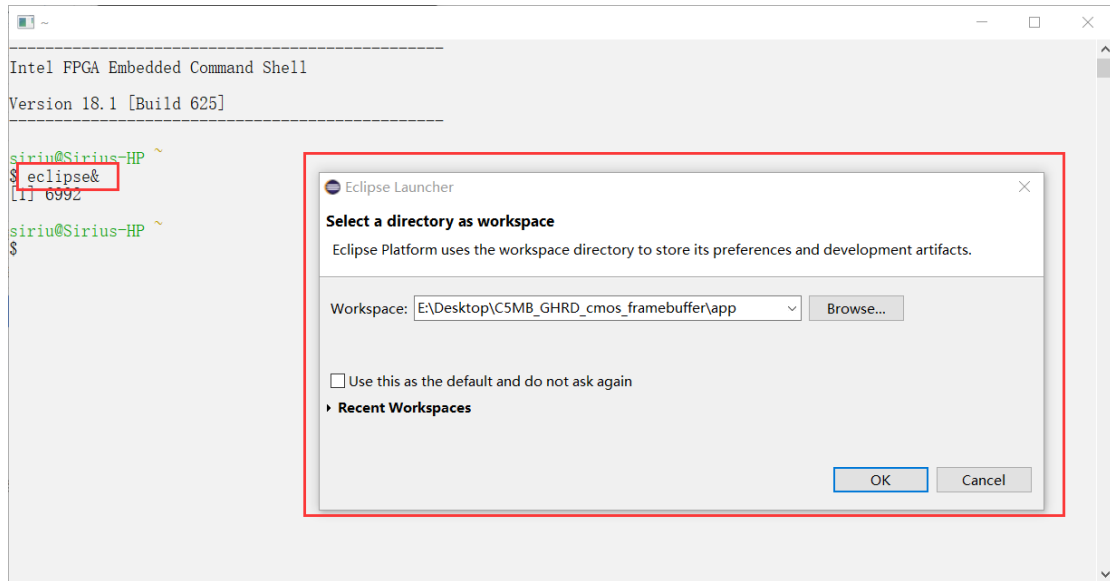
在开始菜单打开 SoC EDS Command Shell



SoC EDS Command Shell 界面

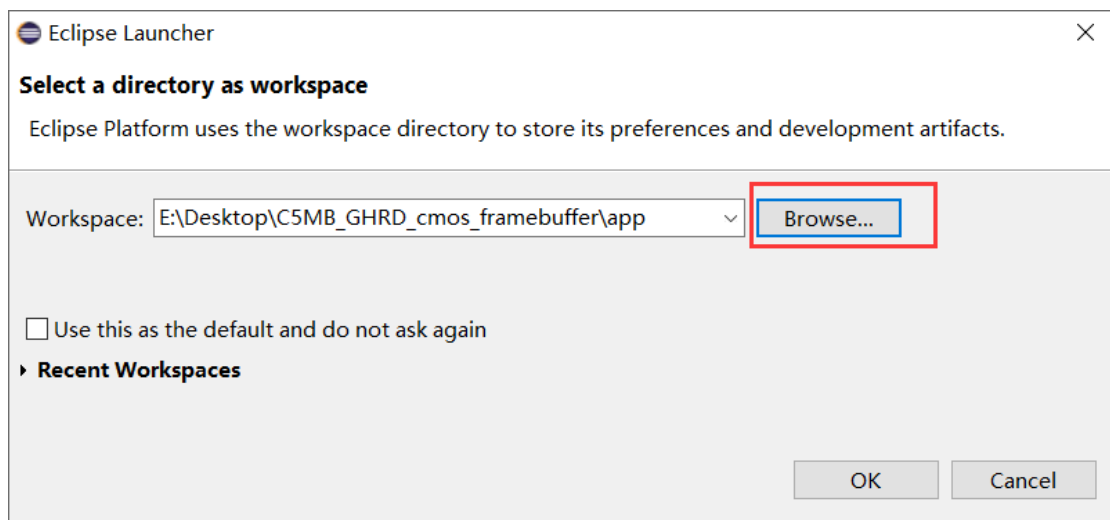
打开 eclipse

在命令行输入 "eclipse&", 回车, 出现以下界面

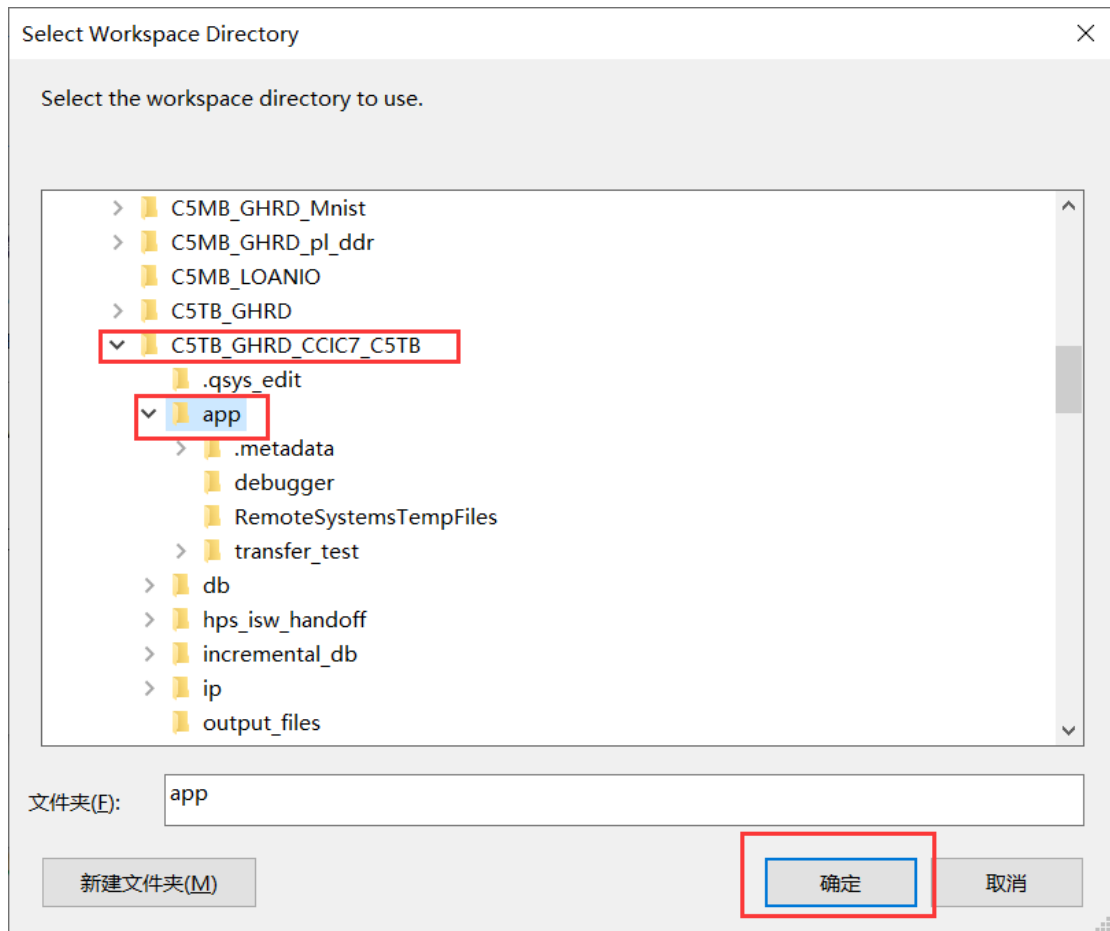


打开示例工程

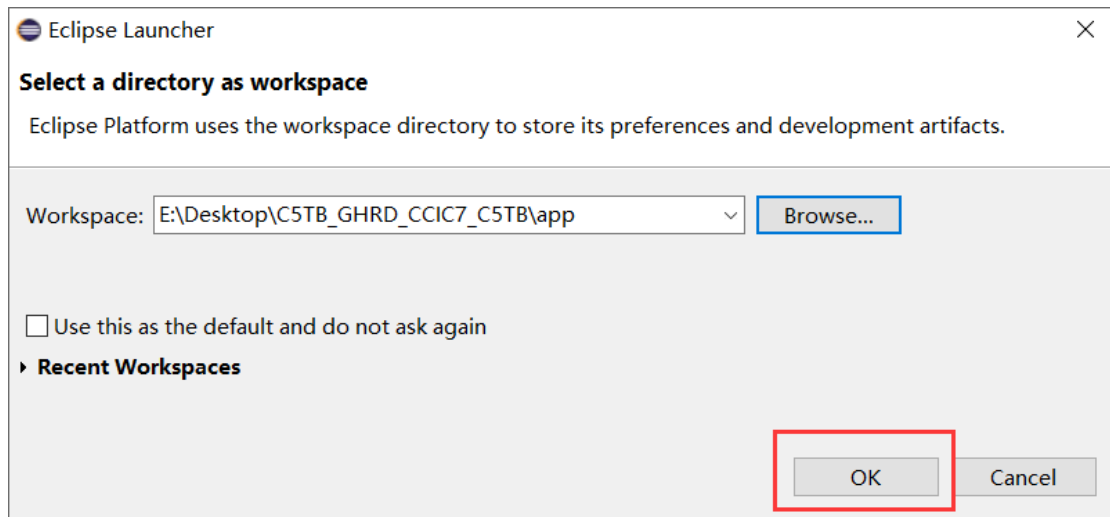
点击 “Browse”



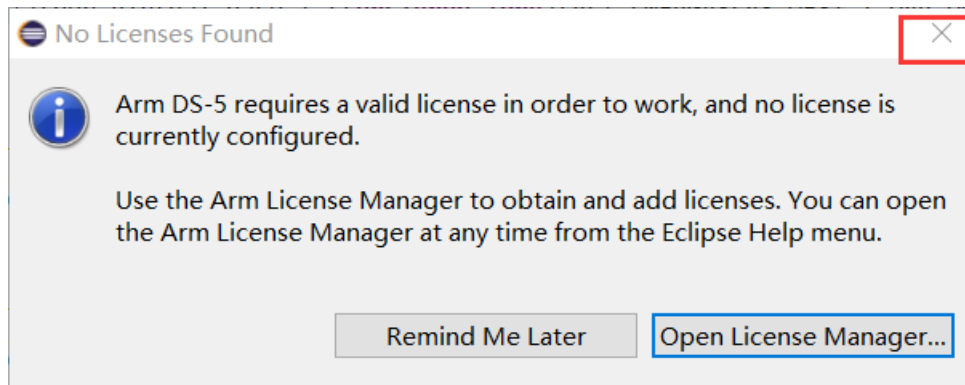
选择工程目录下的 app 文件夹



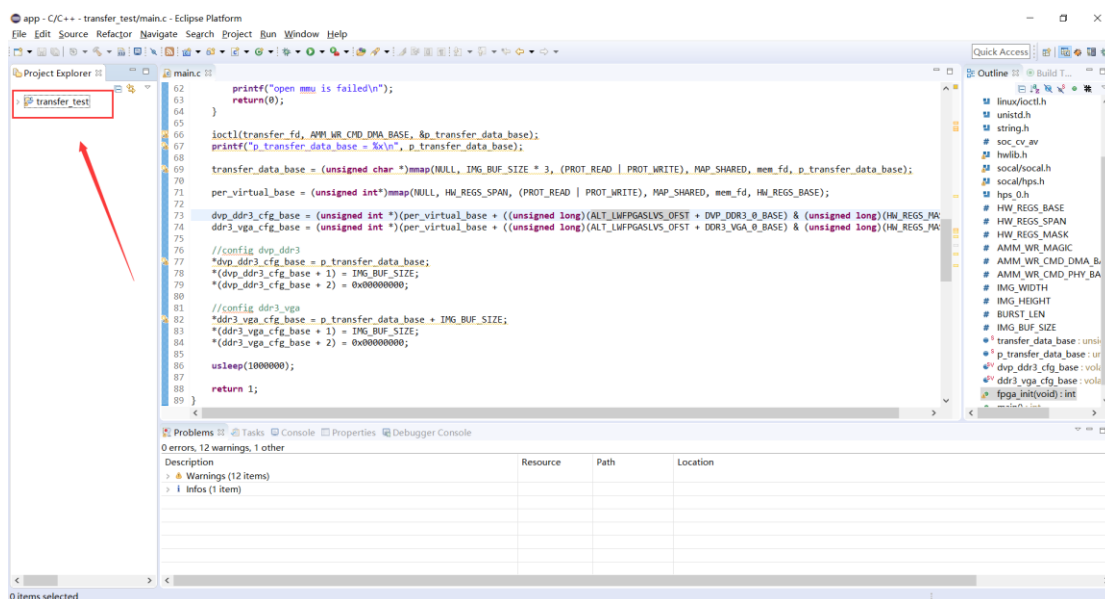
点击 “OK”



出现以下窗口，点击右上角关闭



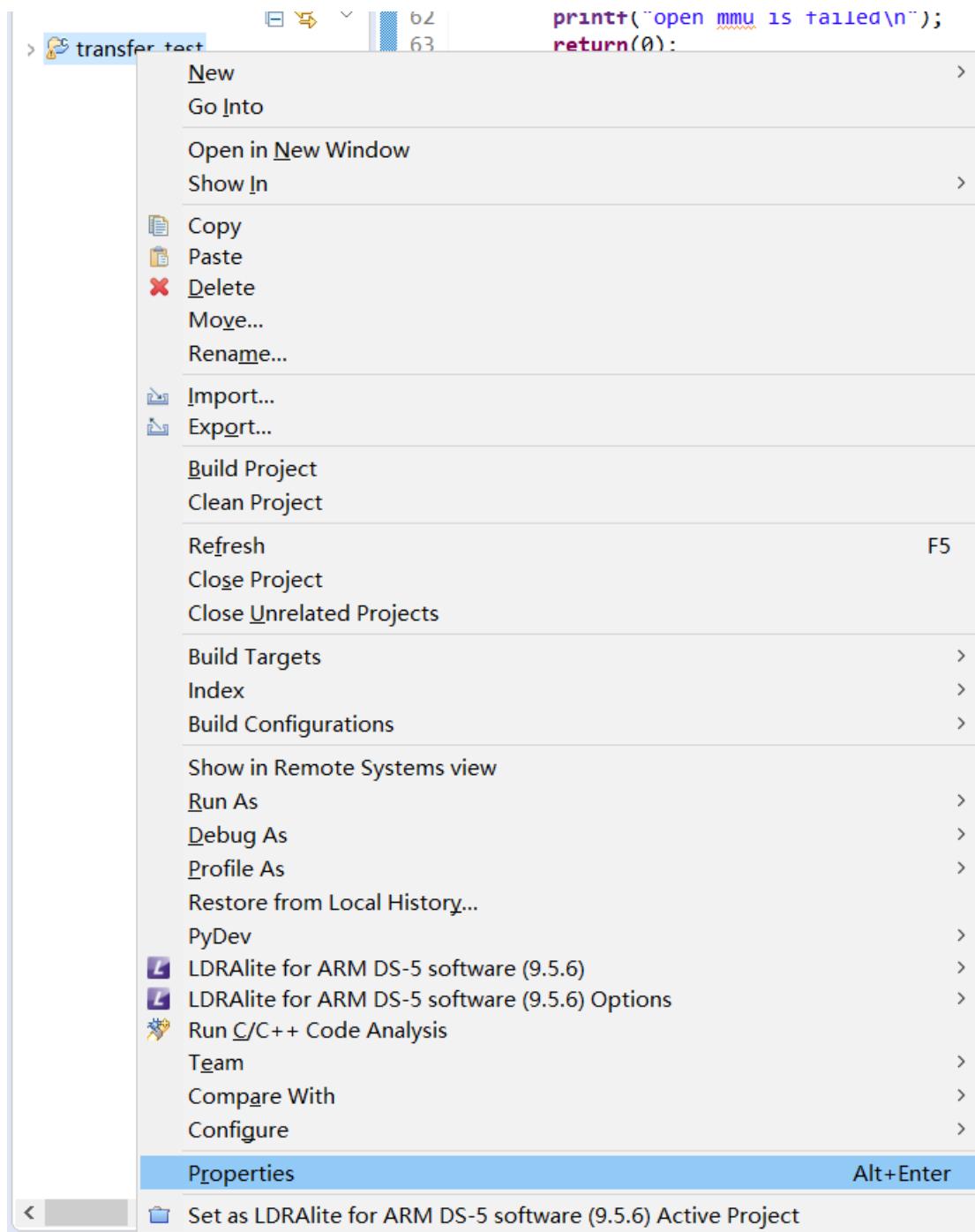
打开过后左侧即为示例工程目录，如下



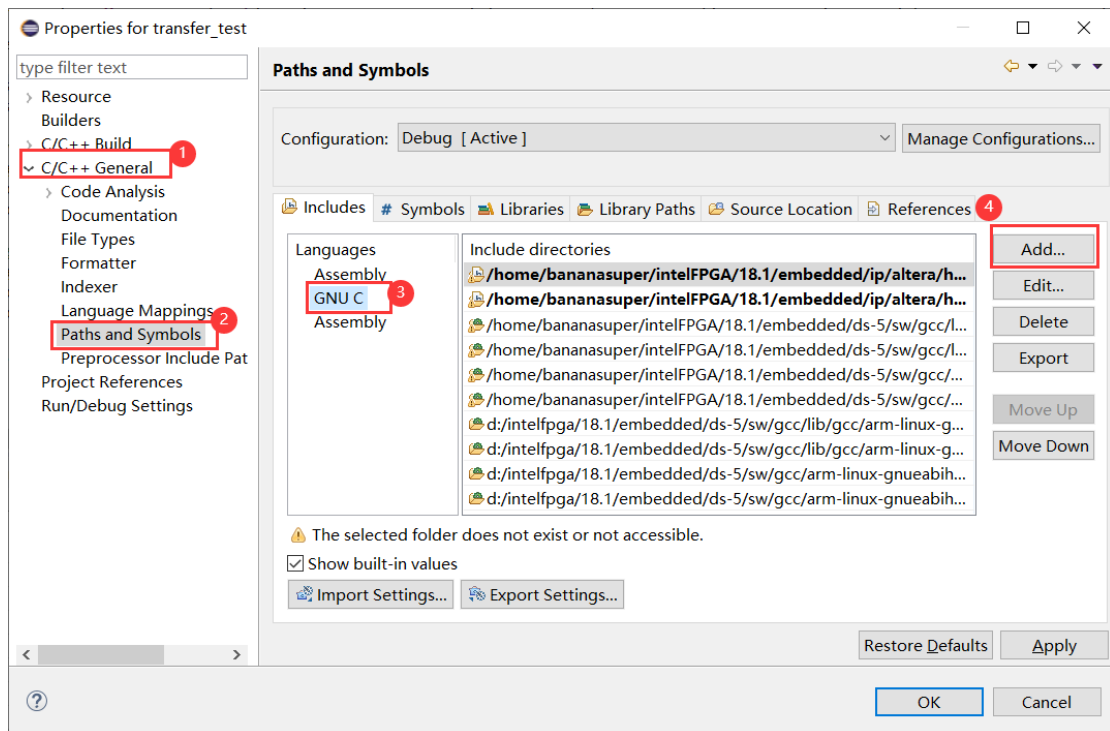
工程名为 **transfer_test**

添加包含文件路径

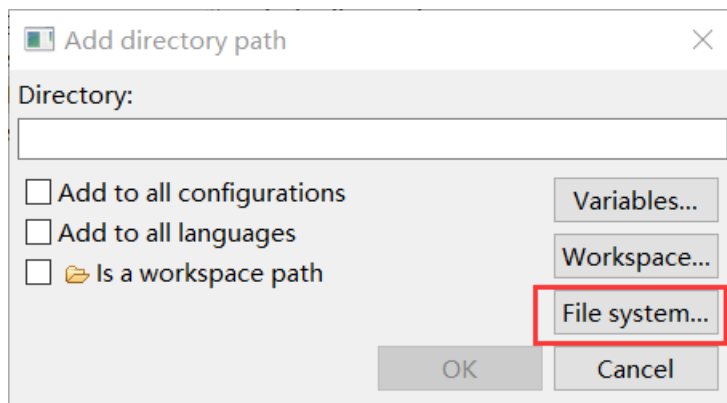
右击工程名，选择“Properties”



依次点击下图位置



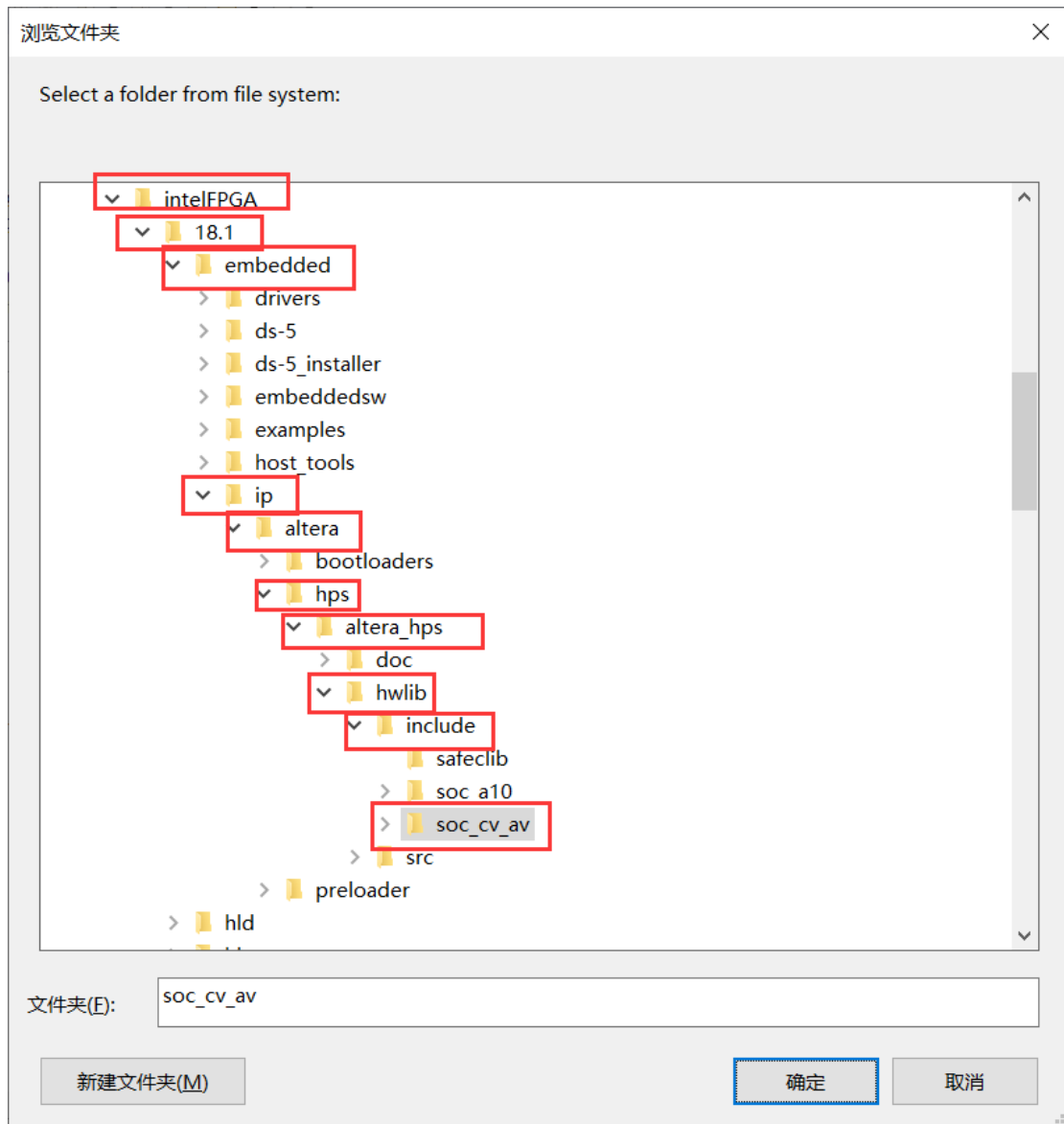
点击 “File system”



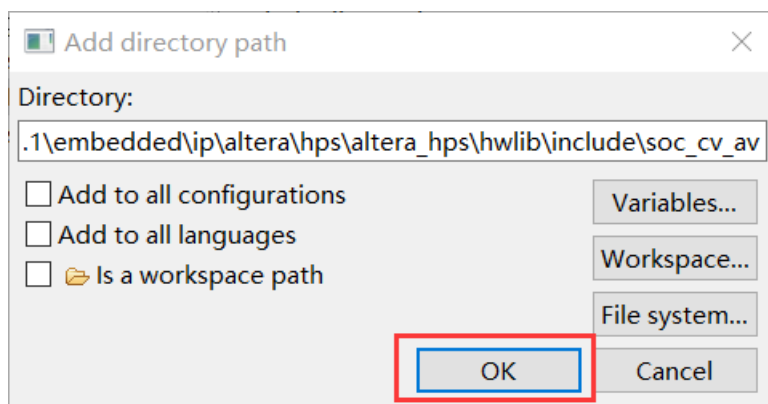
浏览文件夹，选择自己 Quartus 安装路径下的对应文件夹，如我的 quartus 安装在

D:\intelFPGA\18.1 文件夹下，我选择路径为：

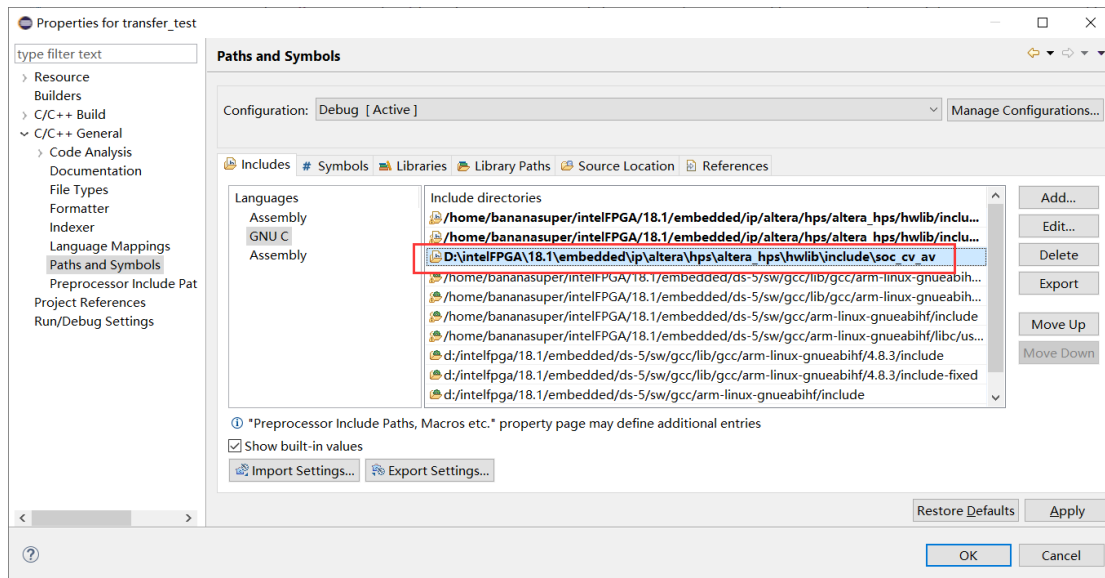
D:\intelFPGA\18.1\embedded\ip\altera\hps\altera_hps\hwlib\include\soc_cv_av



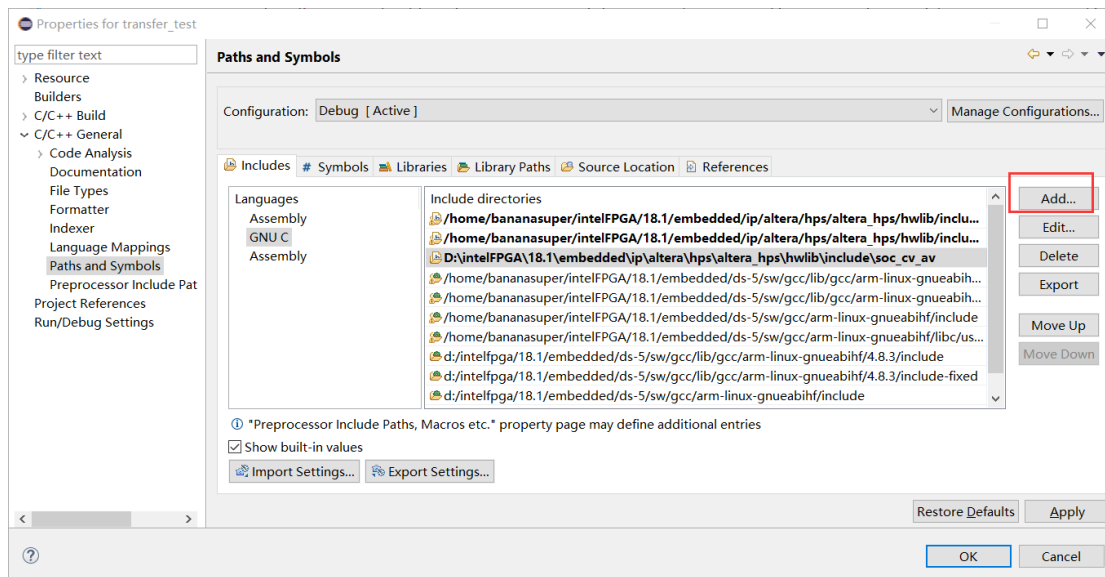
选择完成后点击“OK”（手动选择该路径复制一下）



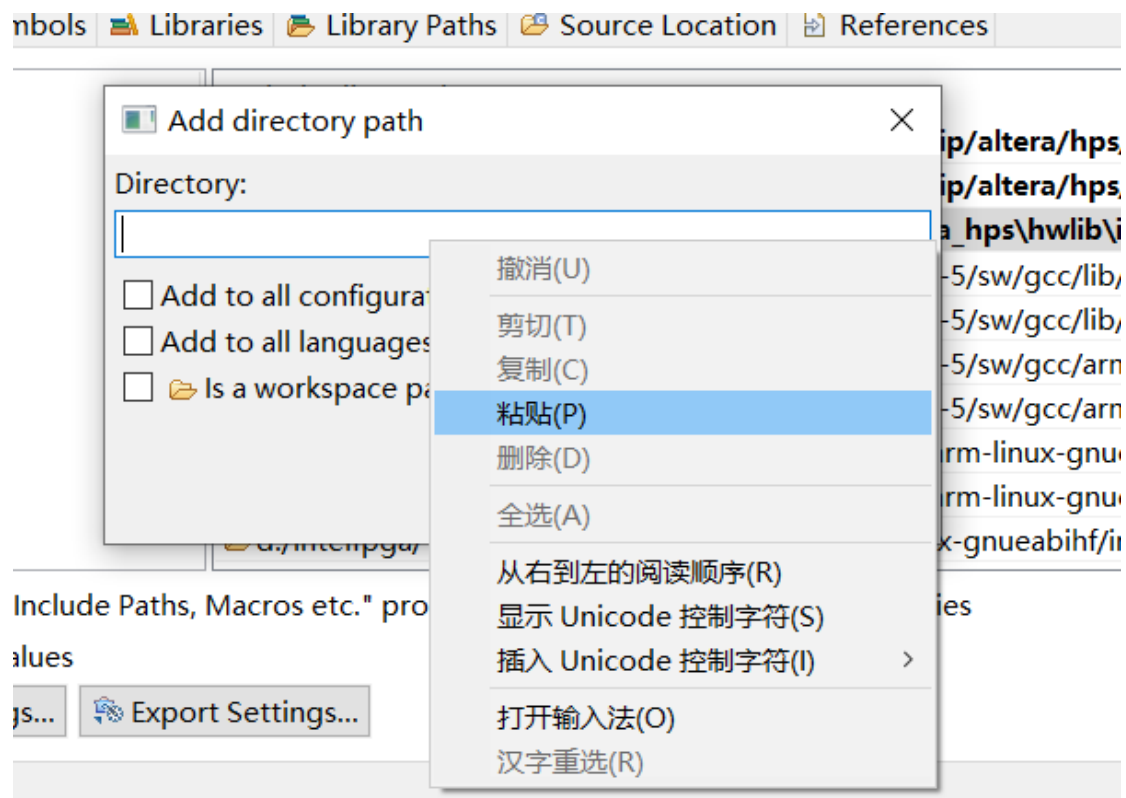
完成后出现以下路径



继续点击 “Add...” 导入



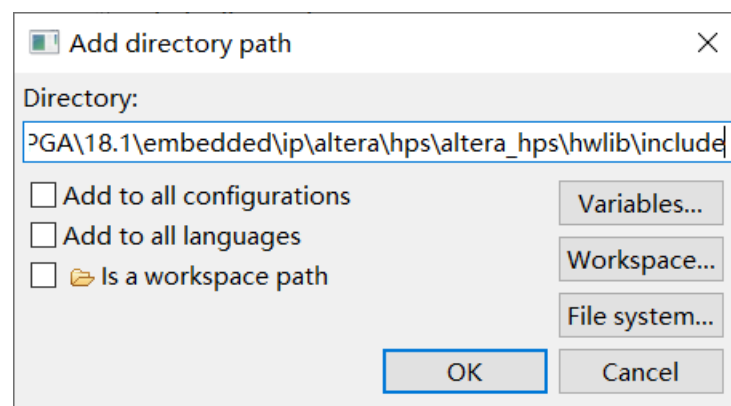
粘贴之前复制的路径



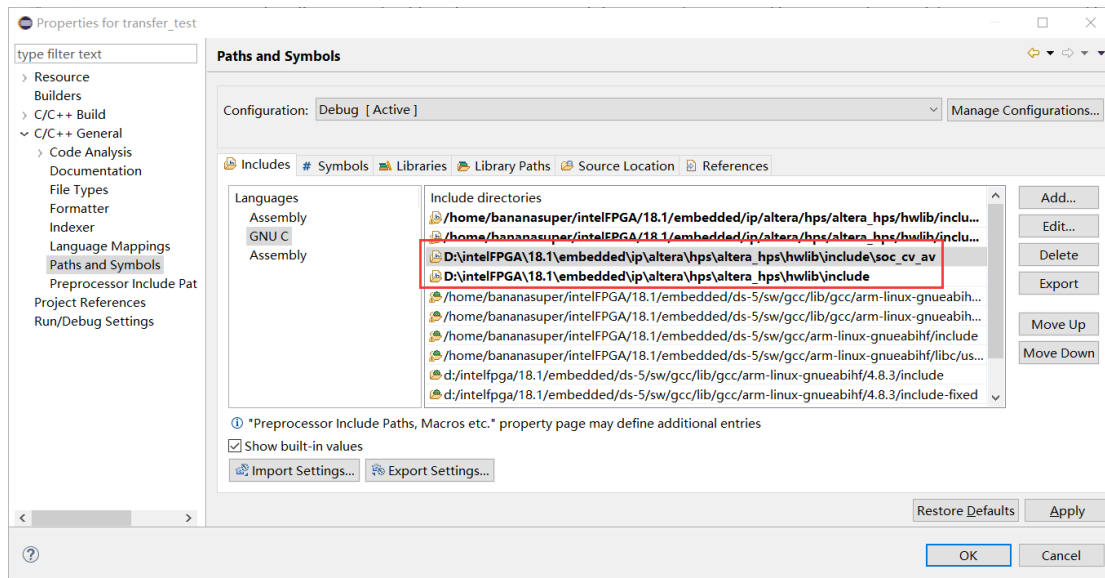
删去之前路径的 \ soc_cv_av

比如我此时的路径为 D:\intelFPGA\18.1\embedded\ip\altera\hps\altera_hps\hwlib\include

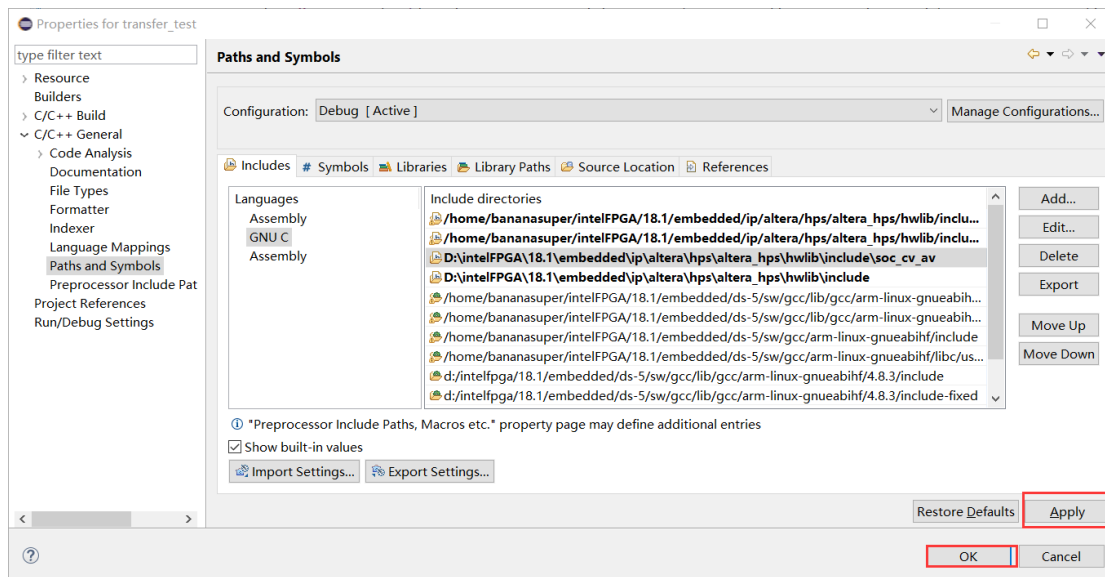
点击 “OK”



添加的路径如下，注意选择自己的安装路径



依次点击 “Apply” “OK”

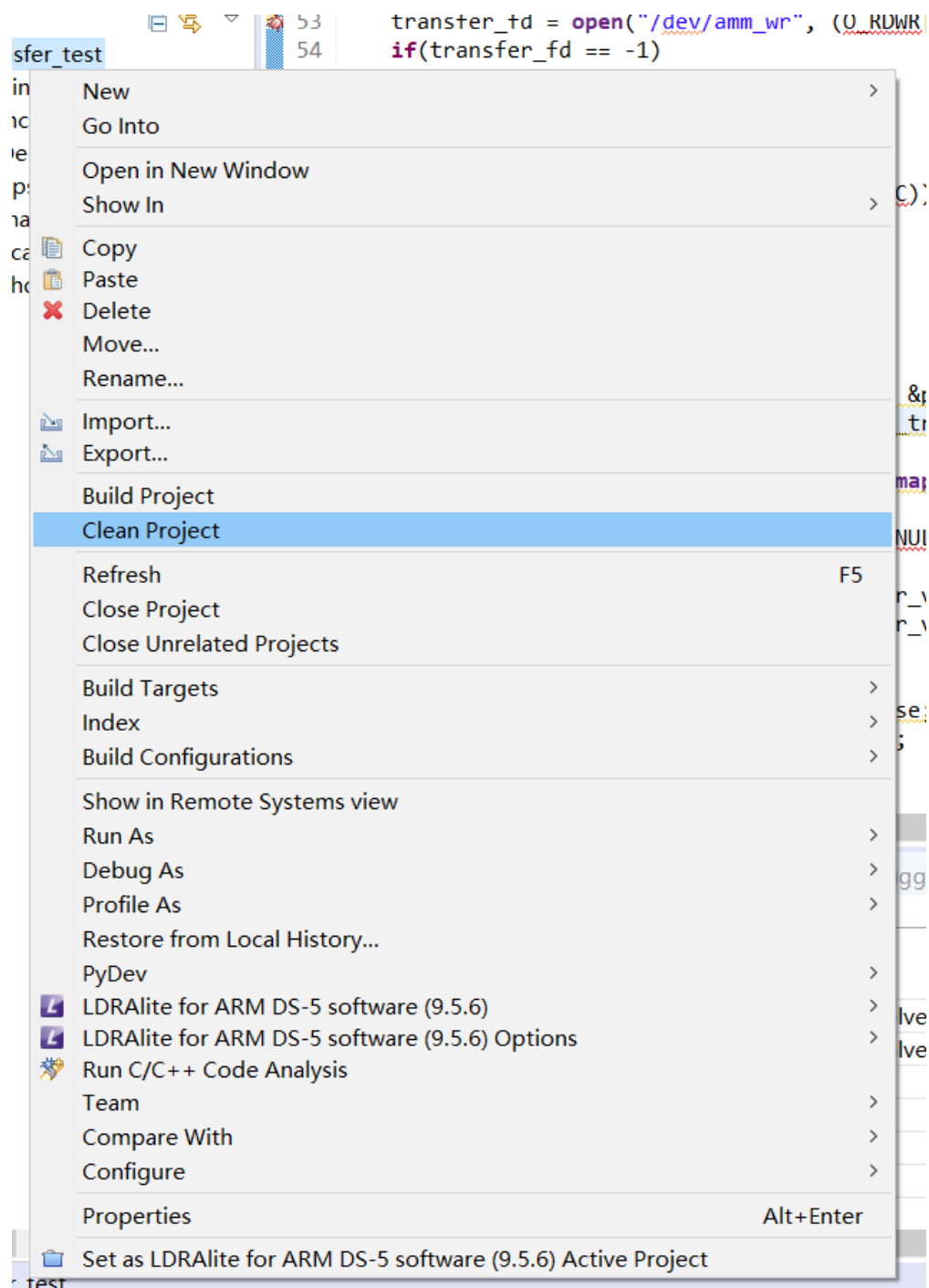


如果工程报错

此时工程可能会报错，（报错是正常现象，**如果没报错可以直接编译**）如下：

Problems 21 errors, 12 warnings, 1 other			
Description	Resource	Path	Location
Symbol 'O_RDWR' could not be resolved	main.c	/transfer_test	line 59
Symbol 'O_SYNC' could not be resolved	main.c	/transfer_test	line 53
Symbol 'O_SYNC' could not be resolved	main.c	/transfer_test	line 59
Symbol 'PROT_READ' could not be resolved	main.c	/transfer_test	line 69
Symbol 'PROT_READ' could not be resolved	main.c	/transfer_test	line 71
Symbol 'PROT_WRITE' could not be resolved	main.c	/transfer_test	line 69
Symbol 'PROT_WRITE' could not be resolved	main.c	/transfer_test	line 71
> Warnings (12 items)			

右击工程，选择“Clean Project”

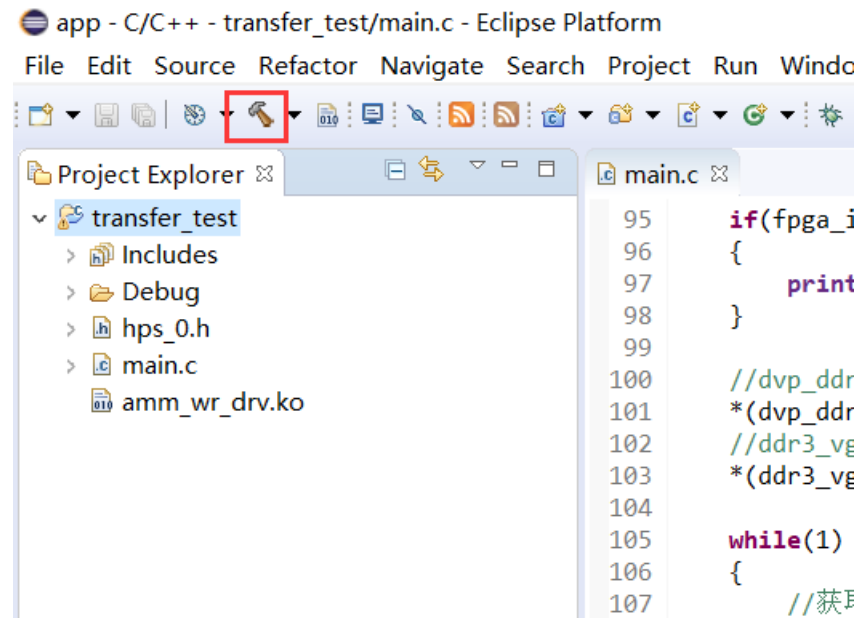


做完这个步骤依然报错，可以选择在工程任意一个报错宏定义中间输入一个空格，然后删除

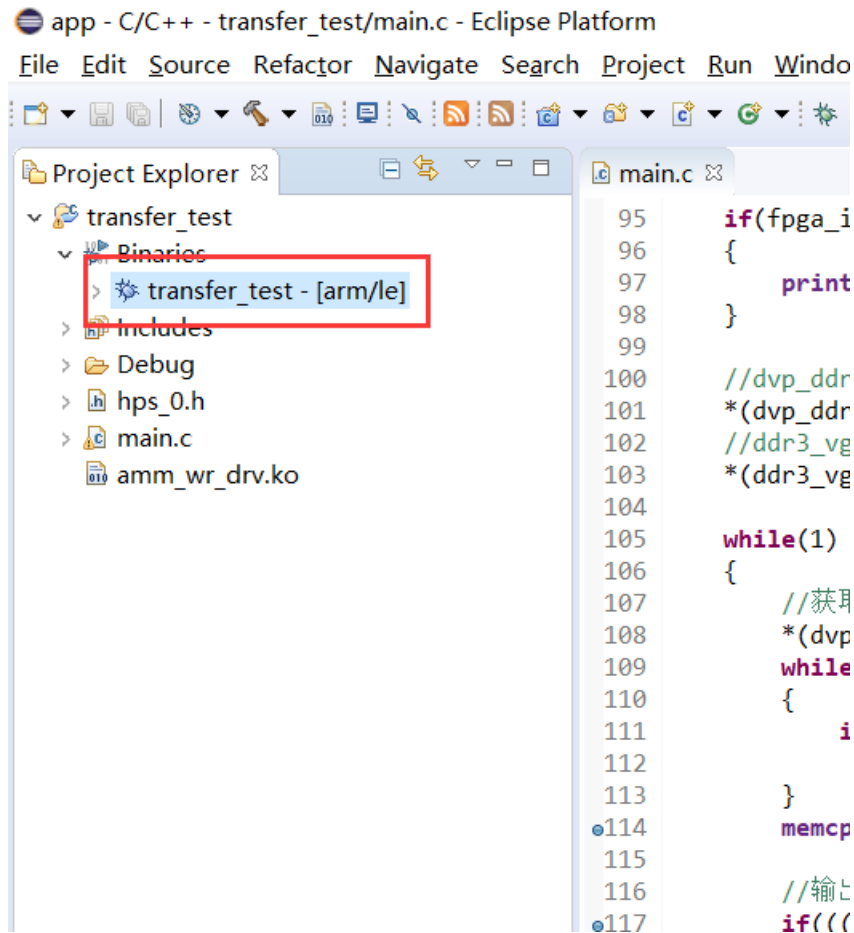
空格，再按 ctrl+s 保存即可

编译示例工程

点击图中锤子图标开始编译工程

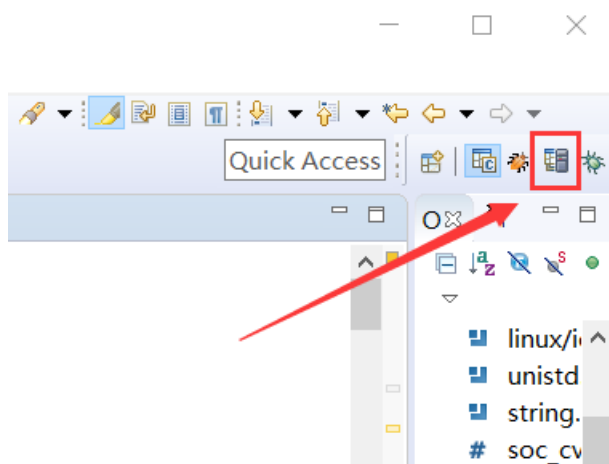


编译完成后在“Binaries”目录下生成如图所示文件

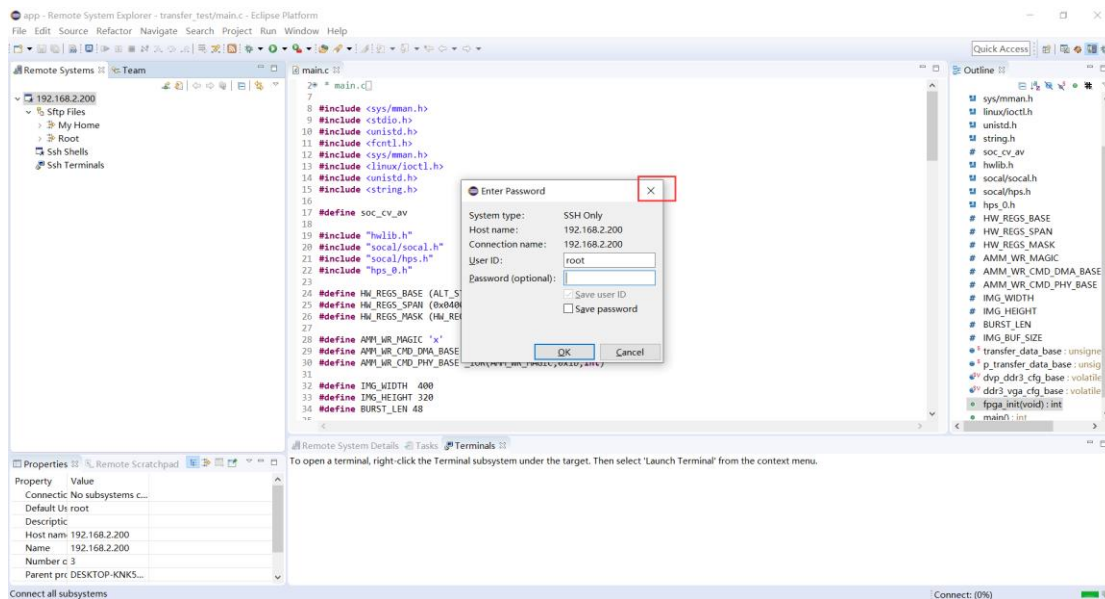


远程登录开发板

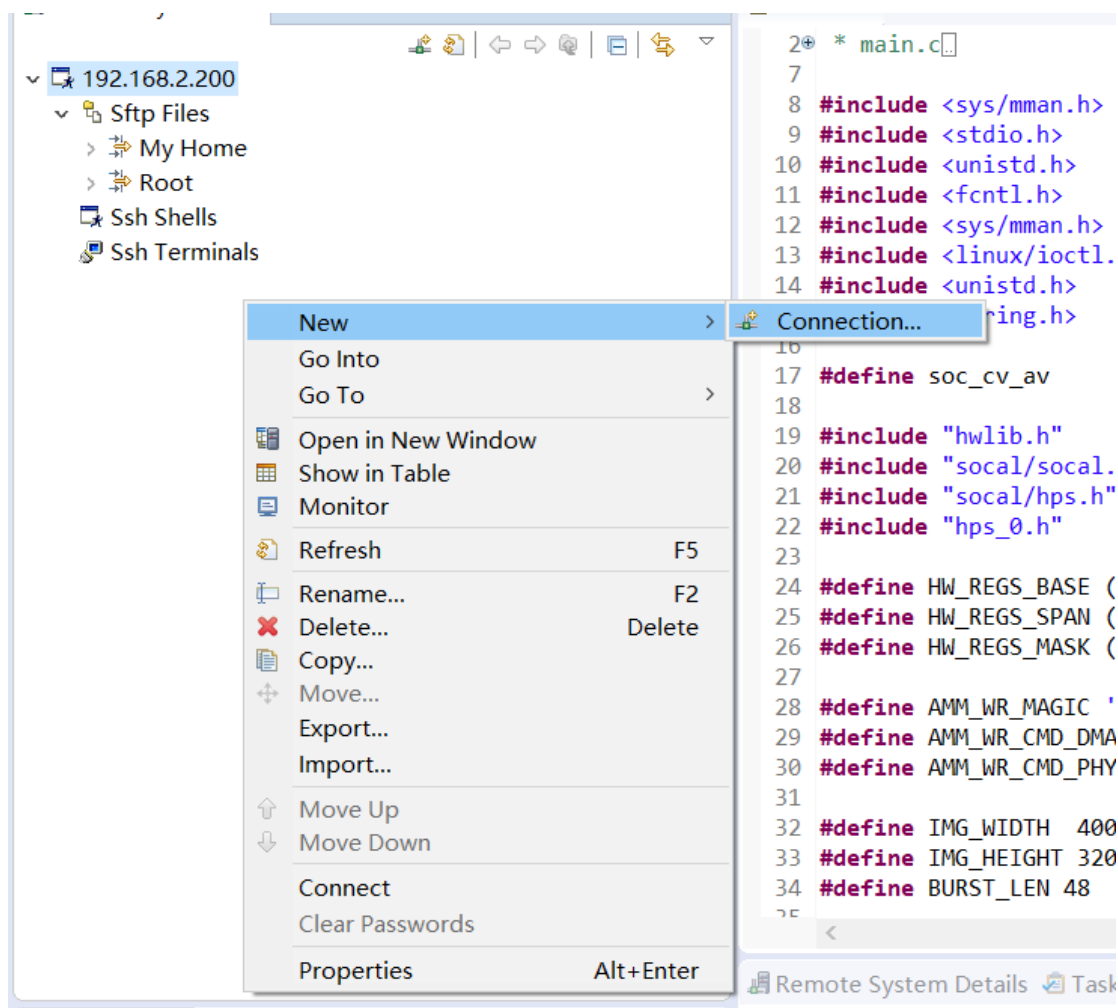
单击右上角如图所示图标，打开远程系统



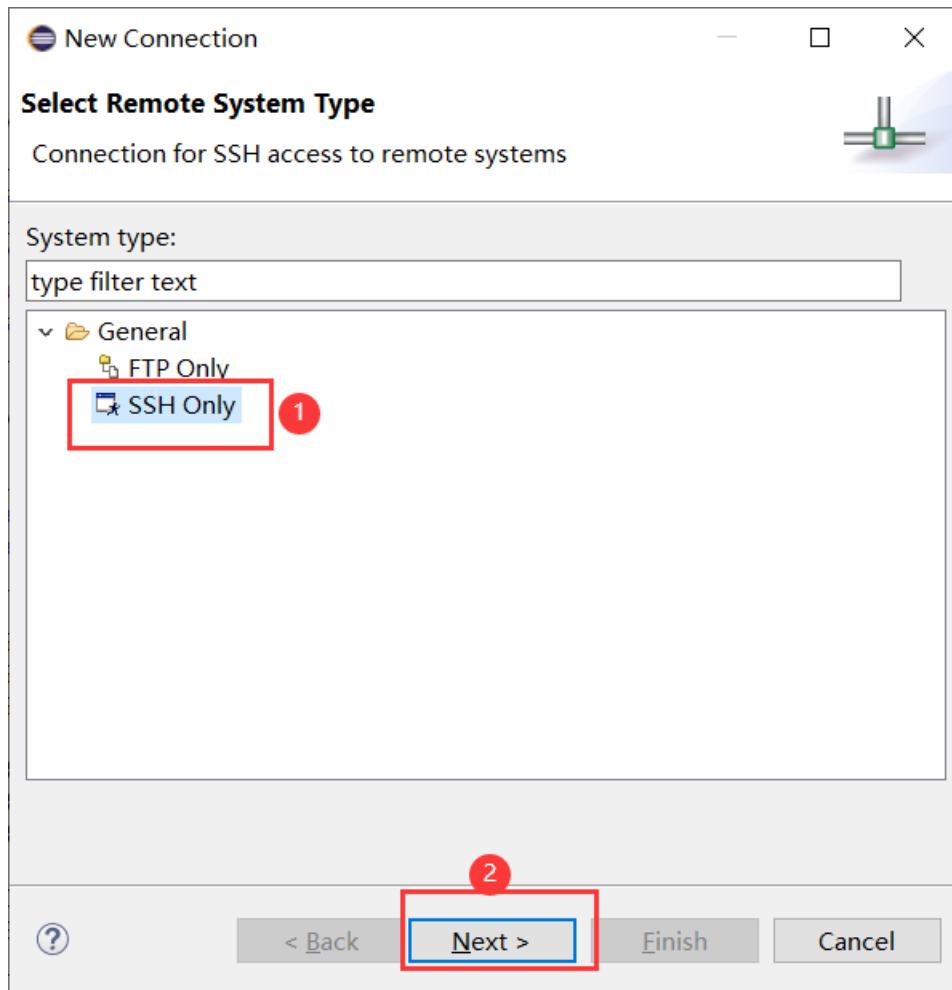
如下，如果出现输入密码的窗口请直接关闭



右击左侧窗口空白区域，出现以下选项，依次选择 “New” -> “Connection”



依次点击 “SSH Only” -> “Next”



输入自己开发板的 IP，比如我的是 172.16.78.11，然后点击 “Finish”

New Connection

Remote SSH Only System Connection

Define connection information

Parent profile: DESKTOP-KNK5F77

Host name: 172.16.78.11

Connection name: 172.16.78.11

Description:

☒ Verify host name

[Configure proxy settings](#)

[?](#) < Back Next > **Finish** Cancel

开发板 IP 在此处查看

文件(G) 开发板(D) 模式(M) 日志(L) 设置(I) 外设(P) 帮助(H)

申请开发板 修复开发板 开发板上电 开发板断电 选择模式 重置开发板 AIEP实验平台

开发板信息

开发板IP	172.16.78.11
开发板用户名	root
开发板密码	LCkaiv
开发板电源	已上电
开发板状态	正常
申请时间	2023-03-23 20:28:50
预计释放时间	2023-05-04 12:28:50
开发模式	常规

FPGA配置

E:/C5TB/output_files/C5TB_top.sof 选择Sof文件

E:/C5TB/output_files/C5TB_top.rbf 选择Rbf文件

上传

日志

[15:09:11] 尝试连接开发板

[15:09:15] 开发板无法连接, 延迟5秒后重试

[15:09:20] 尝试连接开发板

[15:09:24] 开发板无法连接, 延迟5秒后重试

[15:09:29] 尝试连接开发板

[15:09:33] 开发板无法连接, 延迟5秒后重试

[15:09:38] 尝试连接开发板

[15:09:42] 开发板无法连接, 延迟5秒后重试

[15:09:47] 尝试连接开发板

[15:09:51] 开发板无法连接, 延迟5秒后重试

[15:09:56] 尝试连接开发板

X: 937 Y: 46

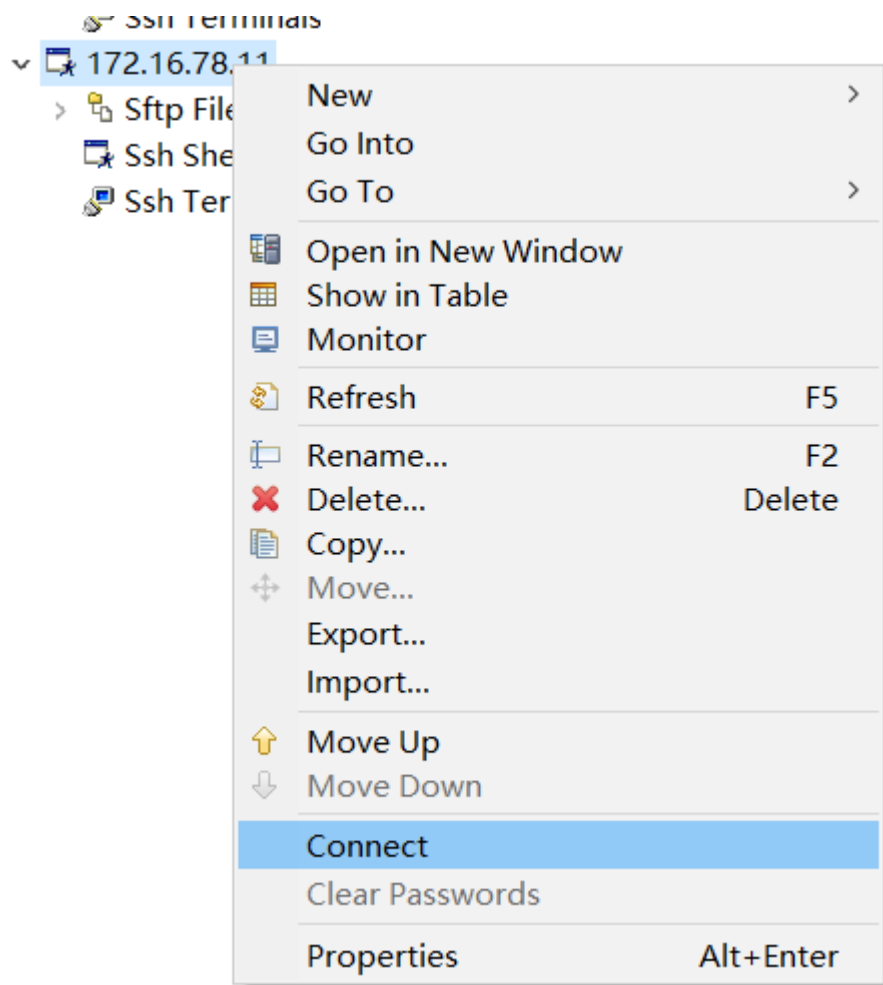
Scale: 100%

延迟: 4ms

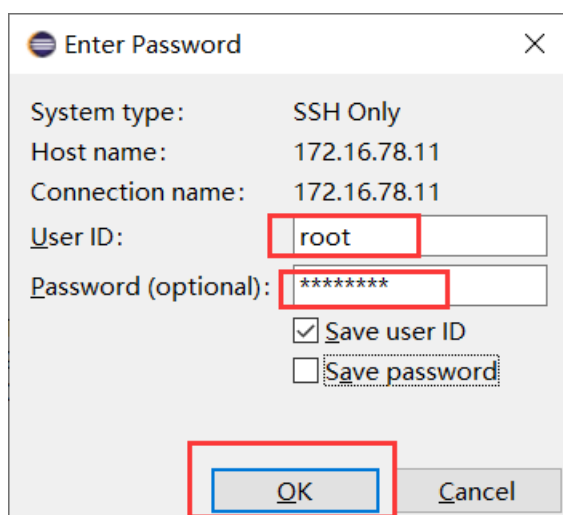
丢包: 0包

丢包率: 0.00%

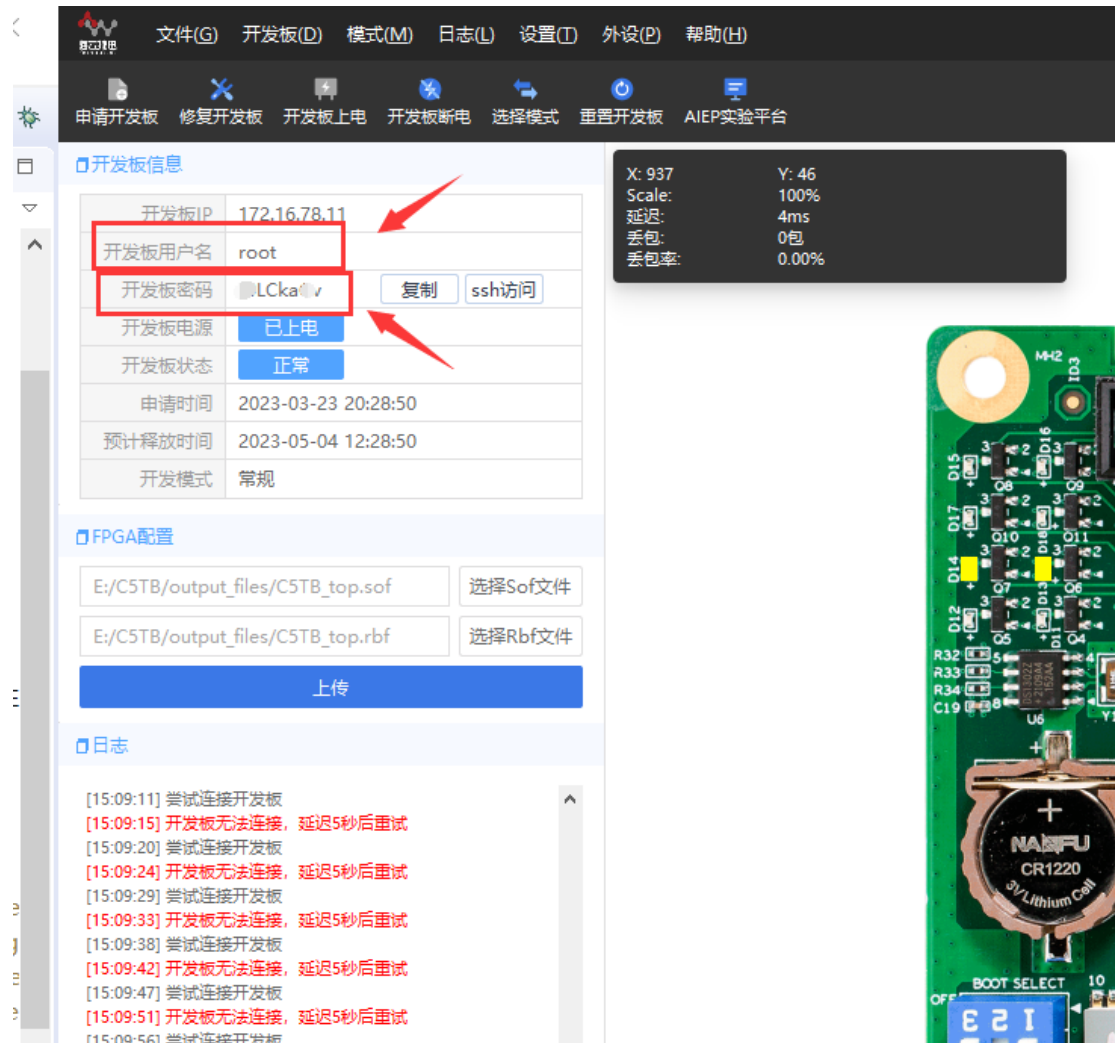
右击 “172.16.78.11”, 选择 “Connect”



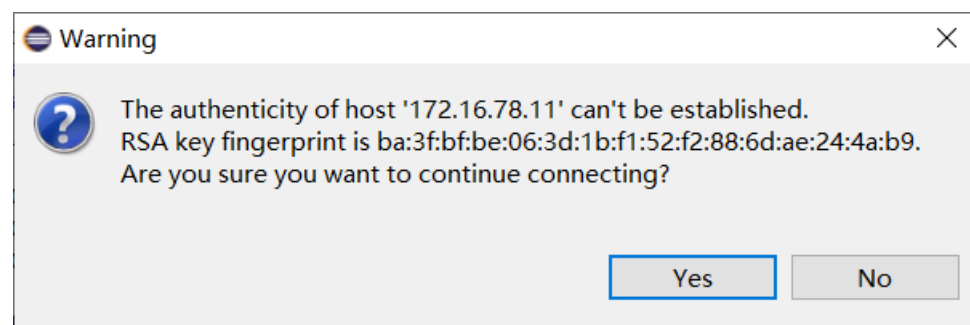
将 User ID 设置为 root, 密码输入开发板的密码



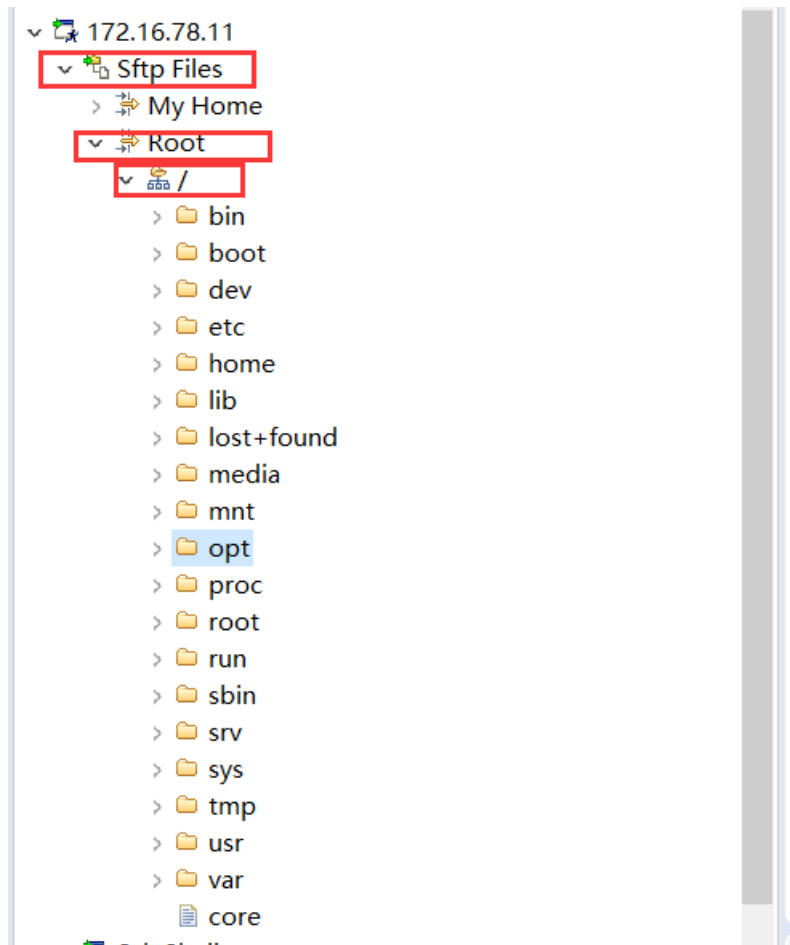
开发板密码在此处查看



选择 "Yes"

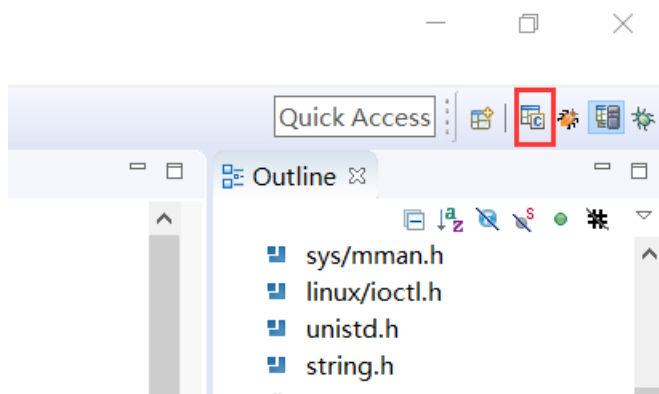


如图可以看到根目录下的信息即为连接成功

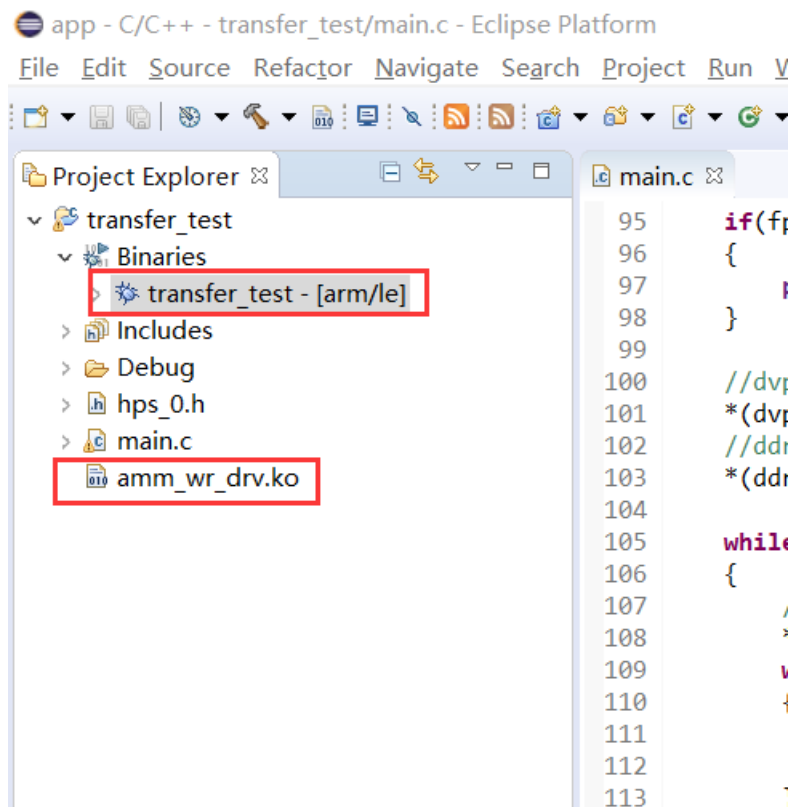


执行示例工程

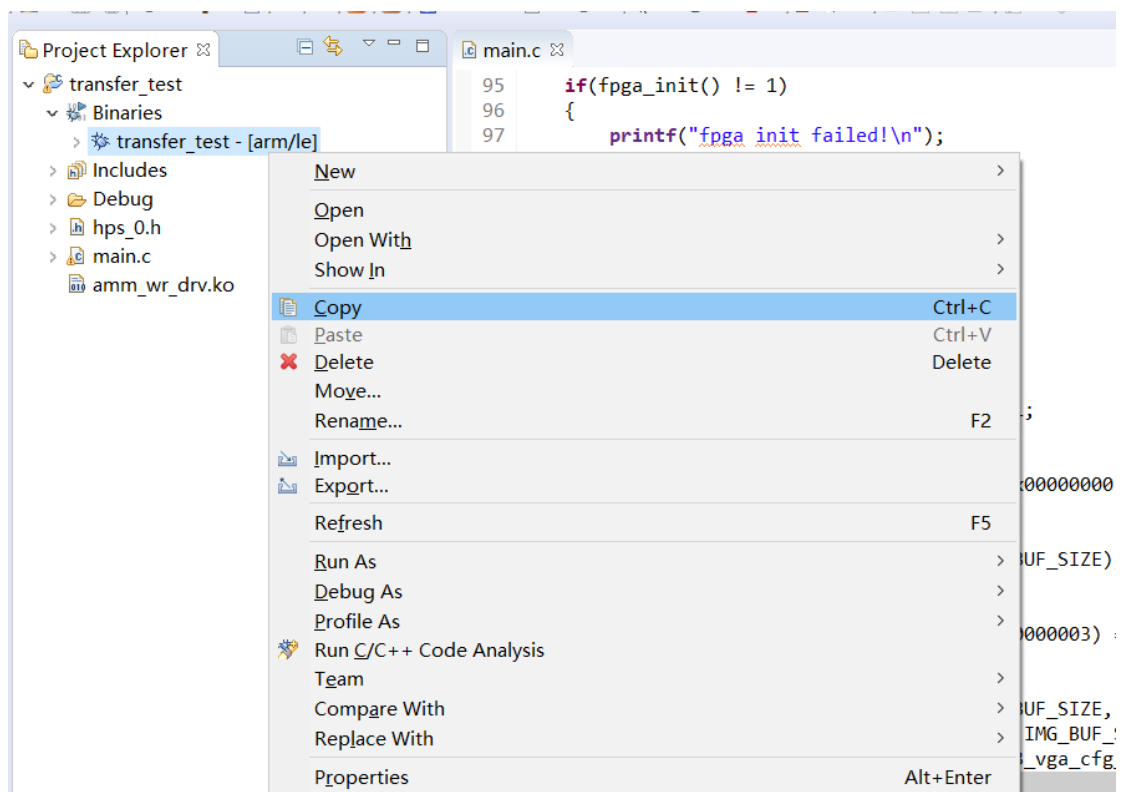
点击右上方如图所示图标，切换回工程目录



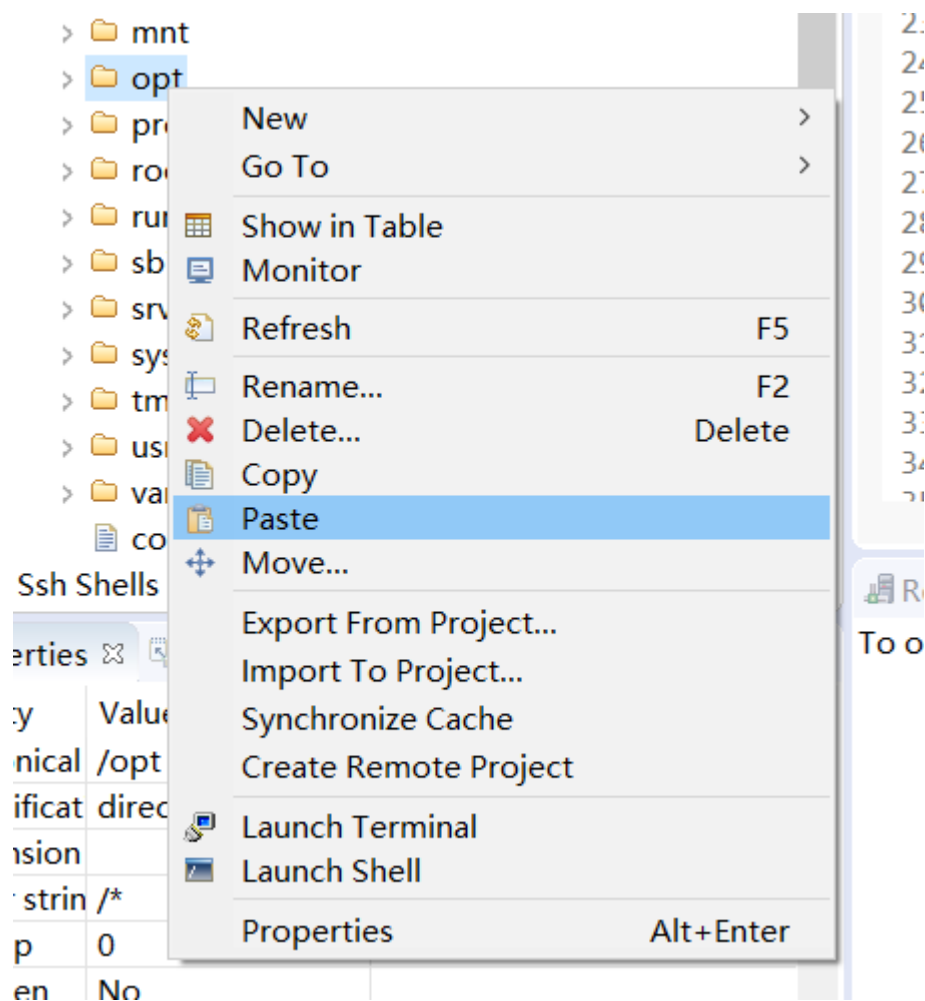
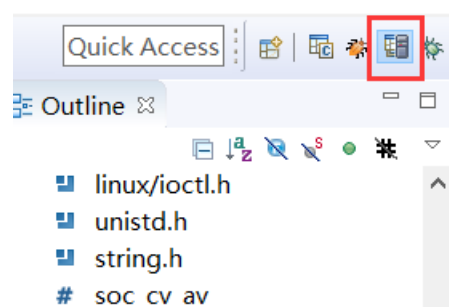
分两次分别复制粘贴下图两个文件到开发板



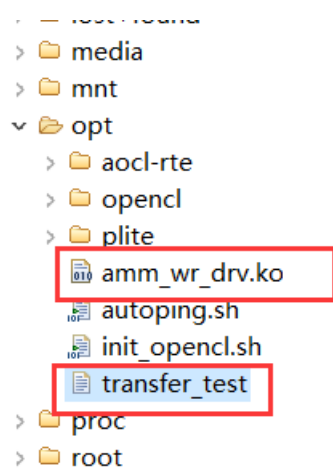
右击选择 “Copy”



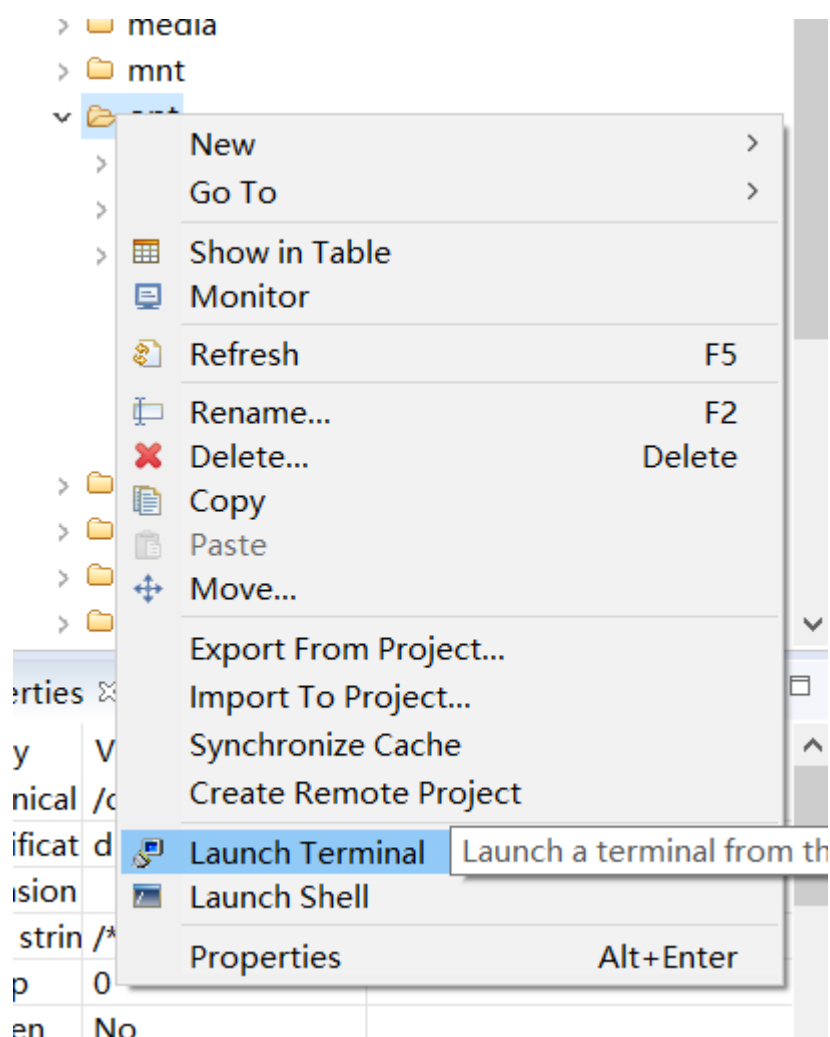
右上角切换回远程系统，右击 “opt” 目录，选择 “Paste”



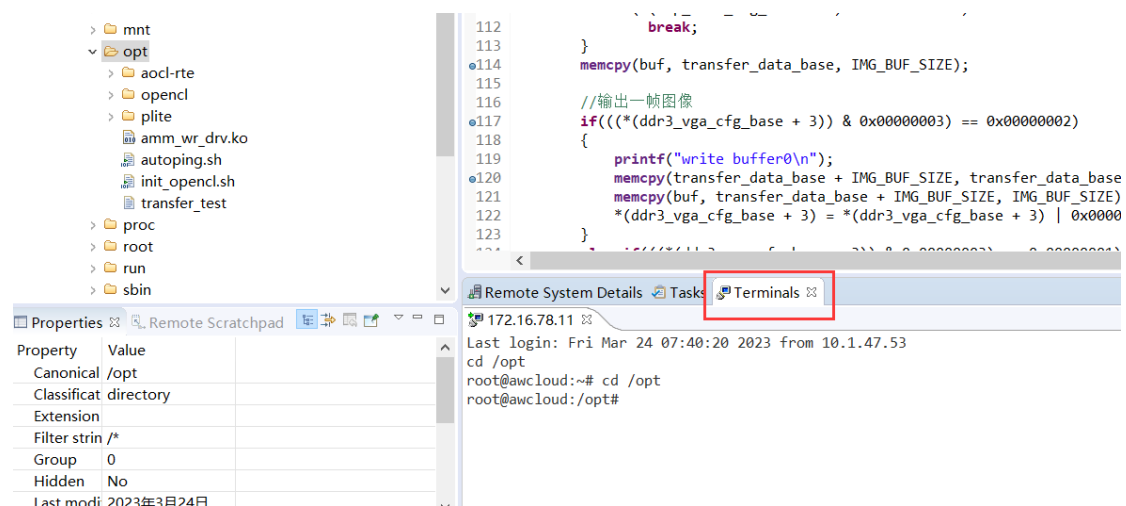
同理再去复制粘贴 `amm_wr_drv.ko`，两次复制粘贴后，如图，`opt` 目录下有了复制的两个文件



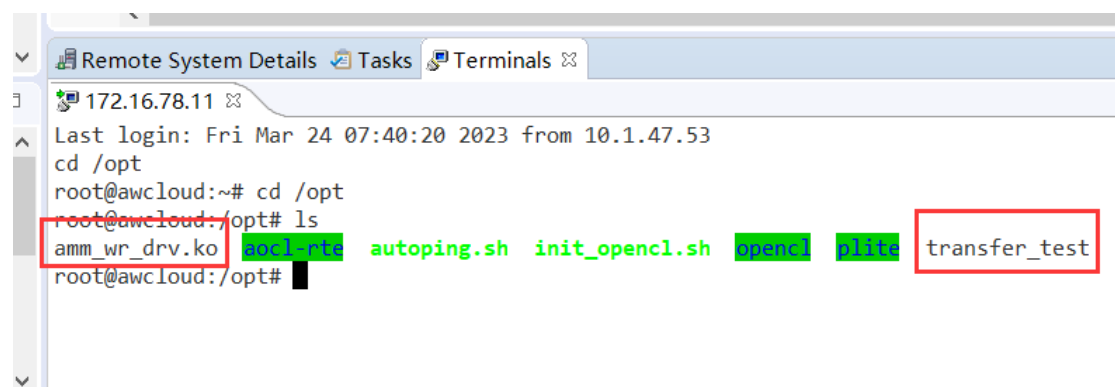
右击“opt”目录，选择“Launch Terminal”



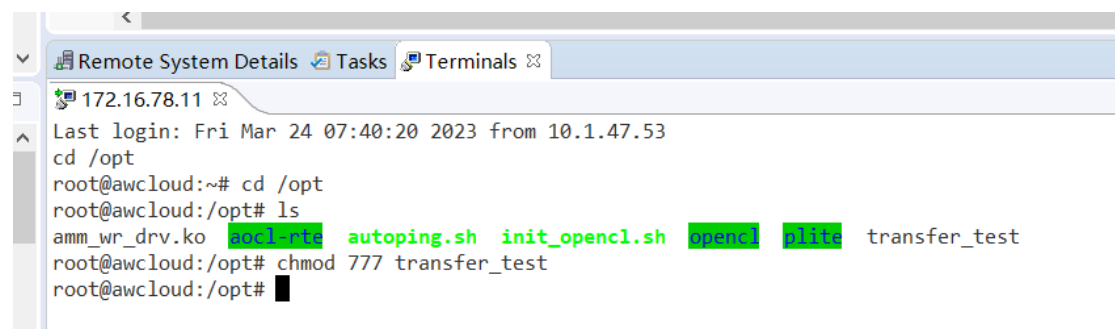
如图打开了终端



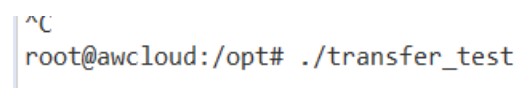
输入 “ls”，回车，查看当前目录下的文件



输入 “chmod 777 transfer_test”，回车，给 transfer_test 可执行权限



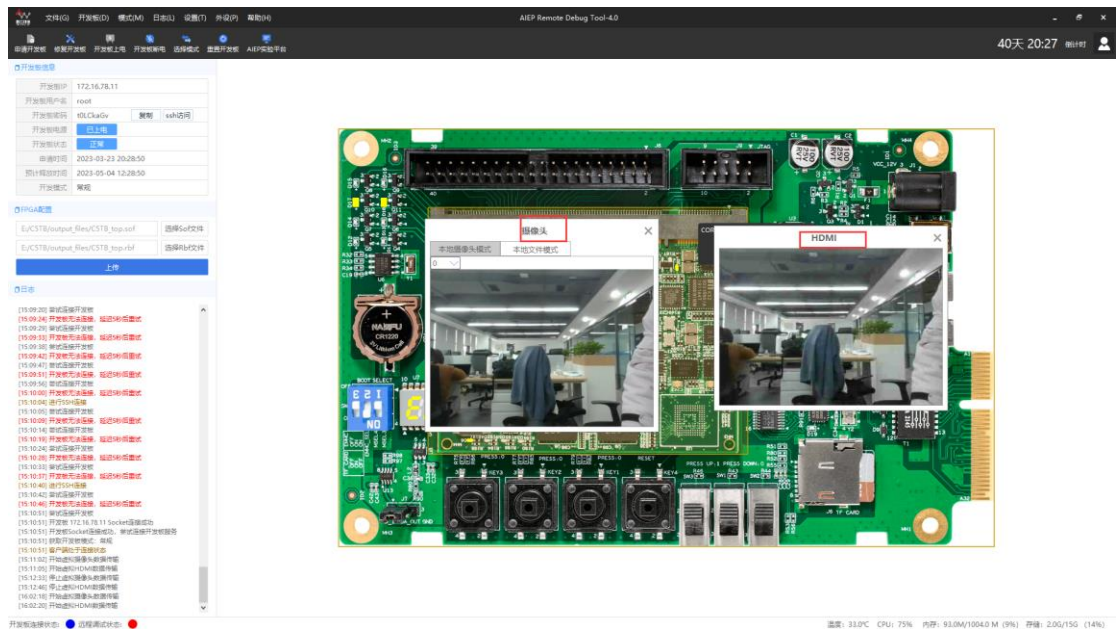
输入 “./ transfer_test”，回车，开始执行 transfer_test



程序开始执行后，在 AIEP 选择 “外设”，可以分别打开摄像头和 HDMI，依次选择打开摄像头和 HDMI



工程实现效果如下图所示，虚拟摄像头采集的图像显示在虚拟 HDMI 显示器上



想要终止程序执行，可以在终端窗口输入 ctrl + c，如图，程序已终止

