

实验一：UNIX V6++运行调制环境的安装与配置

1. 实验目的

- (1) 安装配置 UNIX V6++的运行环境；
- (2) 安装配置 UNIX V6++的调试环境。

2. 实验设备及工具

已安装 Windows 操作系统的 PC 机一台。

3. 预备知识

- (1) gcc 编译
- (2) gdb 调试。

4. 实验准备

请根据自己系统的时机情况，选择下载 32 位或者 64 位的工具包。将工具包解压到本地硬盘的任意位置后，出现如图 1 所示的目录结构。



bochs-2.6	2023/1/25 11:24	文件夹
eclipse	2022/7/25 11:13	文件夹
jdk	2022/7/25 11:16	文件夹
MinGW	2023/1/25 11:23	文件夹
NASM	2022/7/25 11:16	文件夹
oos	2023/1/25 11:25	文件夹

图 1：UNIX V6++工具包目录结构

bochs-2.6: 一款 X86 硬件平台的开源模拟器，它可以在任何编译运行 bochs 的平台上模拟 x86 硬件，包括：I/O 设备，内存和 BIOS。通过改变配置，可以指定使用的 CPU（386，486 或 586），以及内存大小等等。工具包中提供的是免安装的版本。关于 bochs 更多的细节，可搜索百度百科或查询 bochs 官网：<http://bochs.sourceforge.net/>。

eclipse: 免安装的 eclipse 软件包，用于配置 UNIX V6++的调试环境用。

jdk: Java 虚拟机, 运行 eclipse 时需要。

MinGW: 是 Minimalist GNU on Windows 的缩写。它是一个可自由使用和自由发布的 Windows 特定头文件和使用 GNU 工具集导入库的集合, 允许在 GNU/Linux 和 Windows 平台生成本地的 Windows 程序, 而不需要第三方 C 运行时库。这里提供的为免安装版本。关于 MinGW 更多的细节, 可搜索百度百科或查询 MinGW 官网: <http://www.mingw.org/>。

NSAM: 是一个为可移植性与模块化而设计的一个 80x86 的汇编器, 它支持相当多的目标文件格式。这里提供的为免安装版本。关于 NSAM 的更多细节, 可搜索百度百科。请注意, 这里 NASA 需要安装。

oos: 所有 UNIX 代码所在的文件夹。其中, 所有的源代码位于 oos/src。

5. 实验内容

5.1. UNIX V6++运行环境的配置

在解压后的工具包中, “..\oos\targets\UNIXV6++” 目录下, 有 “bochsrc.bxrc” 和 “c.img” 两个文件 (如图 2 所示)。

名称	修改日期	类型	大小
bochscout	2023/1/25 11:25	文本文档	16 KB
bochsrc.bxrc	2023/1/25 11:25	BXRC 文件	2 KB
bochsrc.bxrc	2023/1/25 11:25	快捷方式	1 KB
c	2023/1/25 11:25	光盘映像文件	10,080 KB
debug	2023/1/25 11:25	Windows 批处理文件	1 KB
run	2023/1/25 11:25	Windows 批处理文件	1 KB

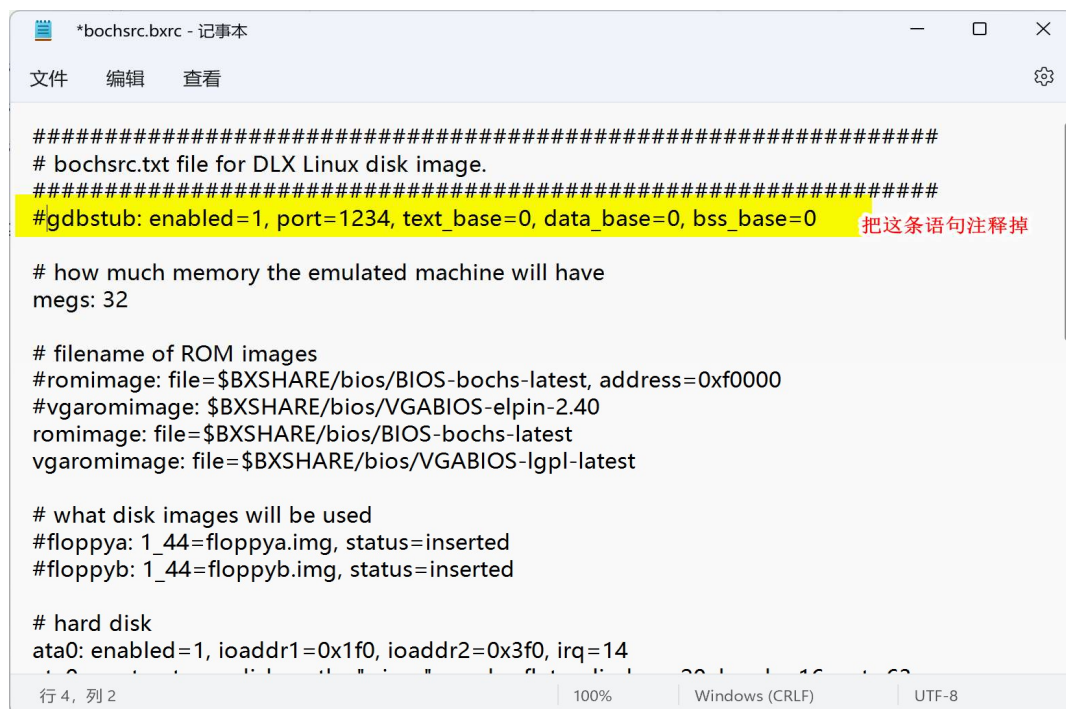
图 2: “UNIX V6++\oos\targets\UNIXV6++” 目录下的文件内容

其中, “bochsrc.bxrc” 是 bochs 的配置文件, 里面配置了 UNIX V6++运行需要的物理环境, 比如 32M 内存等, 图 3 示出了该文件的主要内容。在不影响 UNIX V6++的运行和调试的基础上, 读者可尝试对 “bochsrc.bxrc” 做小的改动。比如, 可将其中 “floppya” 和 “floppyb” 表示两块 UNIX V6++并未使用到的软盘注释掉。

这里需要注意的是: “bochsrc.bxrc” 文件中的第一行:

```
gdbstub: enabled=1, port=1234, text_base=0, data_base=0, bss_base=0
```

请通过加 “#” 将其注释掉, 这一条语句是用来控制 bochs 虚拟机的 gdb 调试功能的, 我们将在下一个实验内容中详细解释。



```
#####
# bochsrc.txt file for DLX Linux disk image.
#####
#gdbstub: enabled=1, port=1234, text_base=0, data_base=0, bss_base=0 把这条语句注释掉

# how much memory the emulated machine will have
megs: 32

# filename of ROM images
#romimage: file=$BXSHARE/bios/BIOS-bochs-latest, address=0xf0000
#vgaromimage: $BXSHARE/bios/VGABIOS-elpin-2.40
romimage: file=$BXSHARE/bios/BIOS-bochs-latest
vgaromimage: file=$BXSHARE/bios/VGABIOS-lgpl-latest

# what disk images will be used
#floppya: 1_44=floppya.img, status=inserted
#floppyb: 1_44=floppyb.img, status=inserted

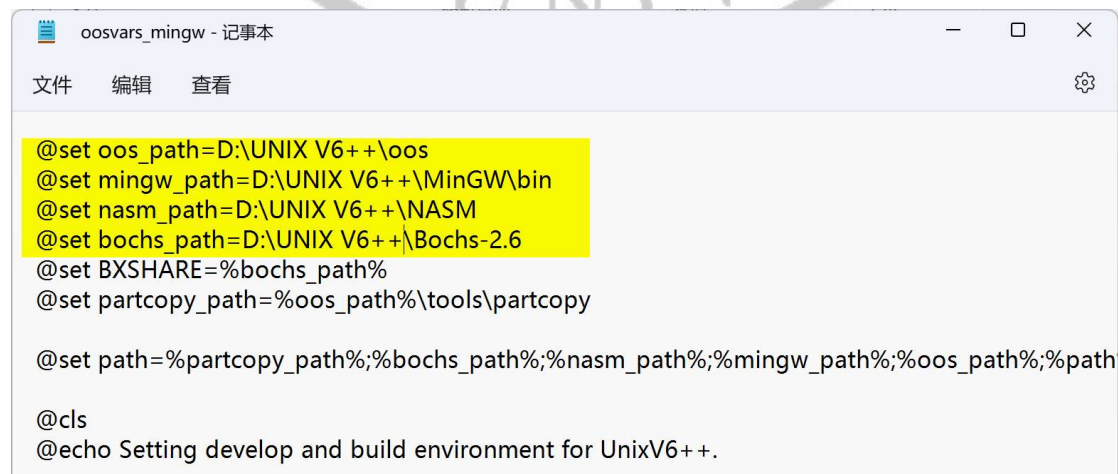
# hard disk
ata0: enabled=1, ioaddr1=0x1f0, ioaddr2=0x3f0, irq=14
```

图 3：“bochsrc.bxrc”文件基本结构

“c.img”是 UNIX V6++的磁盘镜像文件。UNIX V6++文件系统中包含的内容对应于“UNIX V6++\oos\tools\MakeImage\bin\Debug\programs”文件夹下的内容。执行 all 命令后 programs 文件夹下的内容会被写到 UNIX V6++的磁盘文件 c.img。

5.2. 运行 UNIX V6++

为了 UNIX V6++的运行，需要完成一些环境变量的设置，在“UNIX V6++\oos\tools\”目录下，有一个名为“oosvars_mingw.bat”的文件。用文本编辑器打开该文件后，可以看到其中关于 bochs, MinGW 及 NSAM 等工具所在路径的设置。把它们修改成这些文件在你的机器中的正确路径（如图 4 所示）。



```
oosvars_mingw - 记事本

@set oos_path=D:\UNIX V6++\oos
@set mingw_path=D:\UNIX V6++\MinGW\bin
@set nasm_path=D:\UNIX V6++\NASM
@set bochs_path=D:\UNIX V6++\Bochs-2.6
@set BXSHARE=%bochs_path%
@set partcopy_path=%oos_path%\tools\partcopy

@set path=%partcopy_path%;%bochs_path%;%nasm_path%;%mingw_path%;%oos_path%;%path%

@cls
@echo Setting develop and build environment for UnixV6++.
```

图 4：oosvars_mingw.bat 文件内容

完成了上面的各项参数设置之后,运行“UNIX V6++\oos\tools\”目录下的 OOS Command Prompt 快捷方式。运行后弹出如图 5 所示的命令行窗口,在其中输入“run”并回车后,首先启动 bochs 虚拟机,然后启动 UNIX V6++(如图 5 所示)。

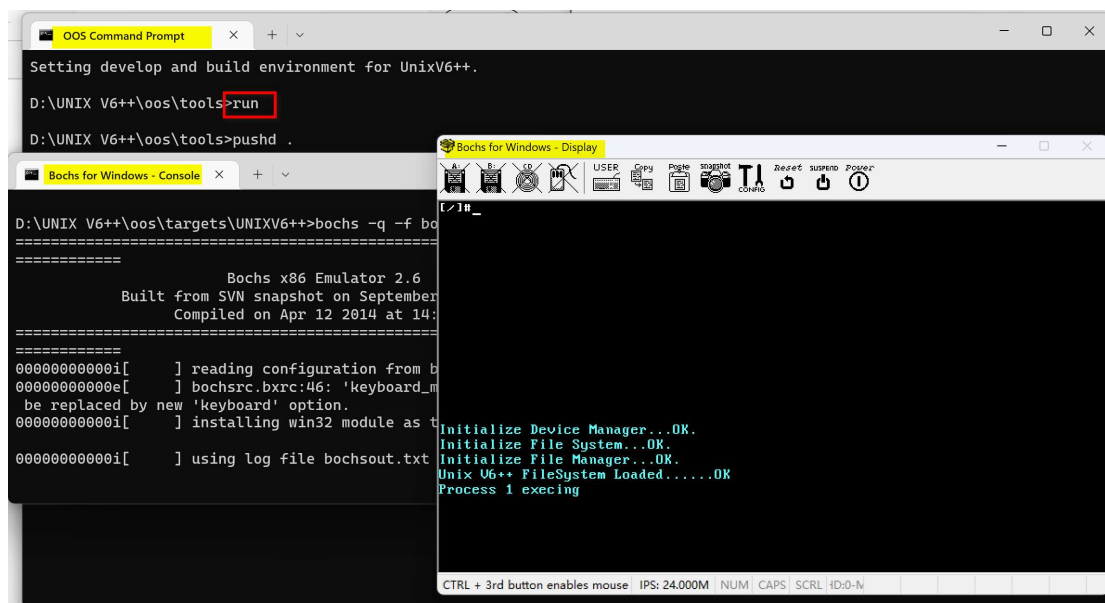


图 5: 通过命令行方式启动 UNIX V6++

在启动的 UNIX V6++终端上输入 shell 命令。UNIX V6++中只实现了很少的几个 shell 命令,可通过图 6 中所示的操作过程查看 UNIX V6++中支持的命令。

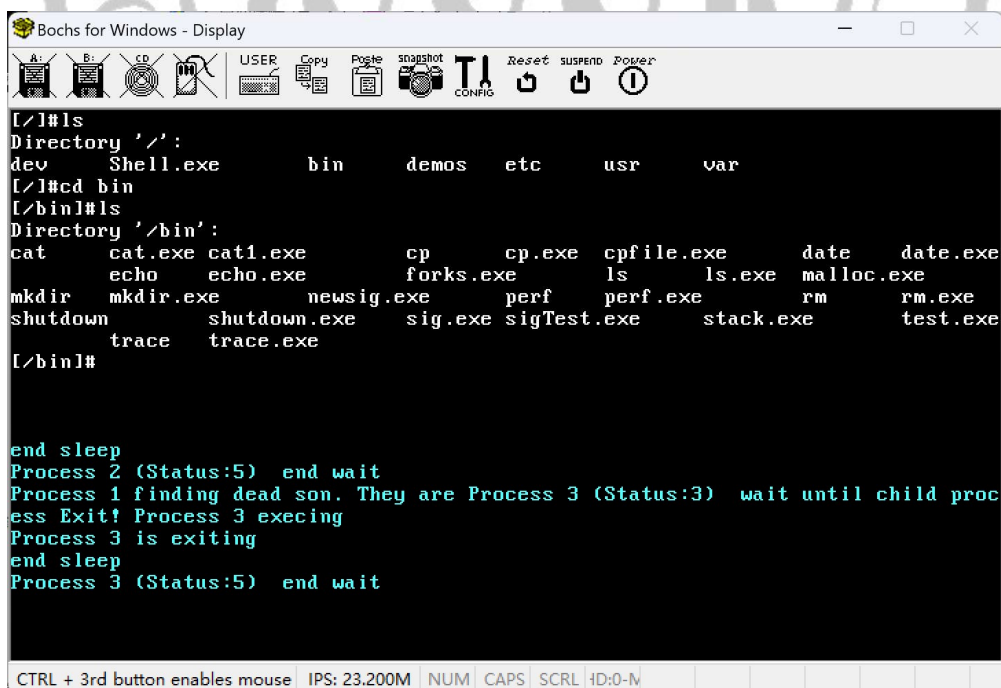


图 6: bochs 虚拟机运行效果

5.3. UNIX V6++调试环境的配置

为了更好的完成对 UNIX V6++ 代码的修改和重新编译, 我们需要一个 gdb 的调试环境。这里我们详细介绍这个 gdb 调试环境的建立过程。

(1) 开启 bochs 虚拟机的调试选项

在实验一中, 我们曾经提到过 “bochsrc.bxrc” 文件中的第一行:

`gdbstub: enabled=1, port=1234, text_base=0, data_base=0, bss_base=0`

是需要被注释掉的, 表示这个时候的 bochs 虚拟机是不支持 gdb 调试功能的, 所以首先需要将这一句的注释去掉 (如图 7 所示)。

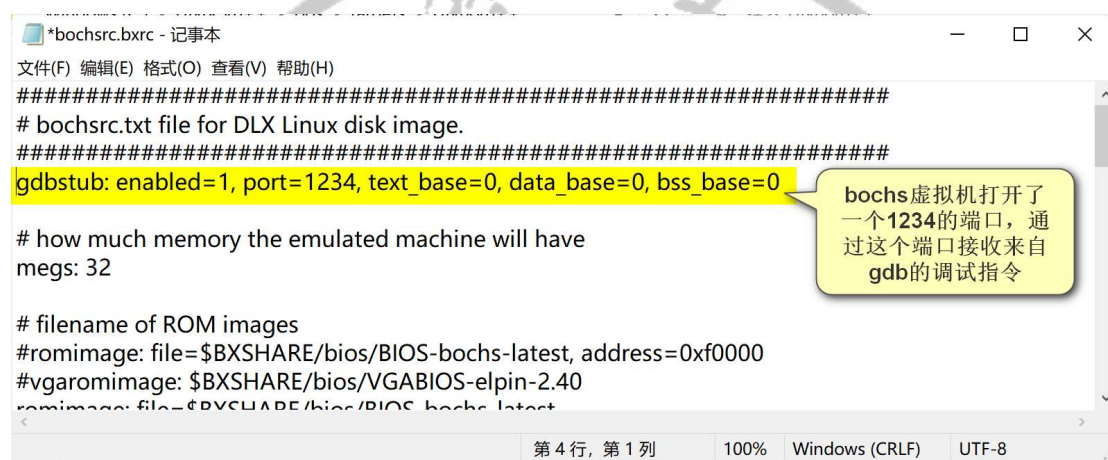


图 7: 开启 gdbstub 注释项

这个时候, 我们再次运行 UNIX V6++ (注意, 如果这时候系统弹出如图 8 所示的对话框, 请在勾选 “专用网络” 之后点击 “允许访问”), 此时, 将看到如 9 所示的运行效果。虚拟机没有再继续运行下去, 而是暂停运行后, 等待来自 gdb 的调试指令。UNIX V6++ 的运行也暂停了。

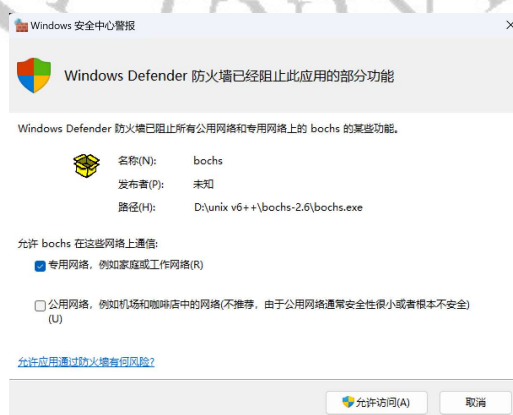


图 8: 关闭 Windows 防火墙

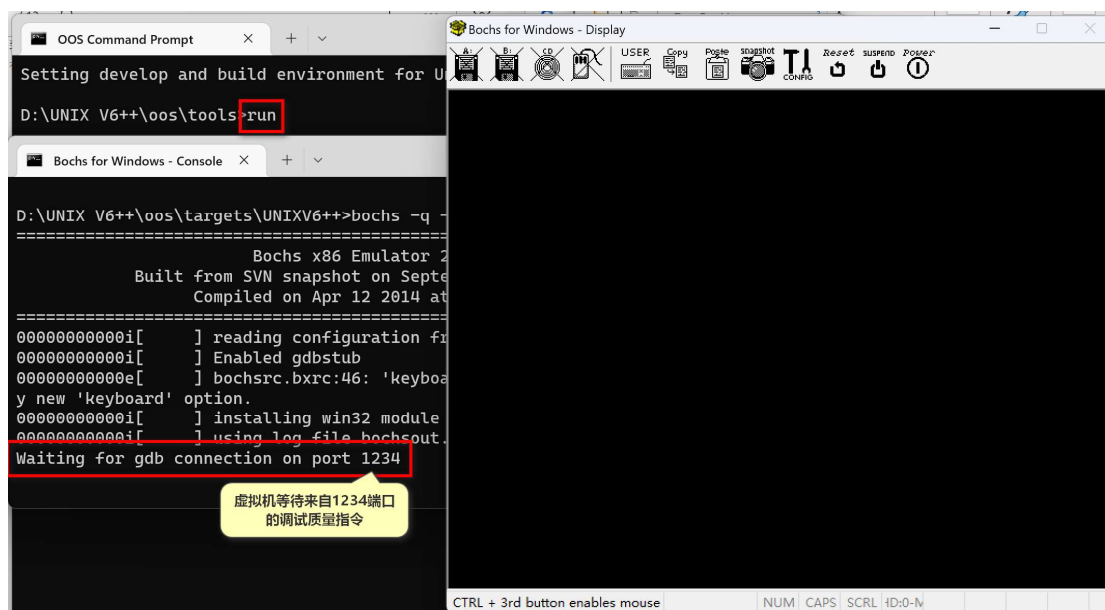


图 9：等待调试指令的虚拟机和 UNIX V6++

至此，熟悉 gdb 调试的同学，可以打开另外一个终端，通过 gdb 调试指令来控制 UNIX V6++ 的指令，完成所有的调试任务。但是，由于 gdb 的调试比较复杂，难于掌握，为了使用的方便，我们将继续以下步骤，以实现利用 eclipse 提供的图形化的前端来完成 UNIX V6++ 的远程调试。

（2）调试相关的系统环境变量设置

为了支持 UNIX V6++ 的调试，需要在 windows 操作系统中设置环境变量，具体的操作过程为：（1）在 Windows 系统中找到“用户环境变量”设置，（2）选择其中的“path”变量，通过编辑该变量，将 MinGW 和 NASM 的路径添加在最前面。具体流程如图 10 所示。

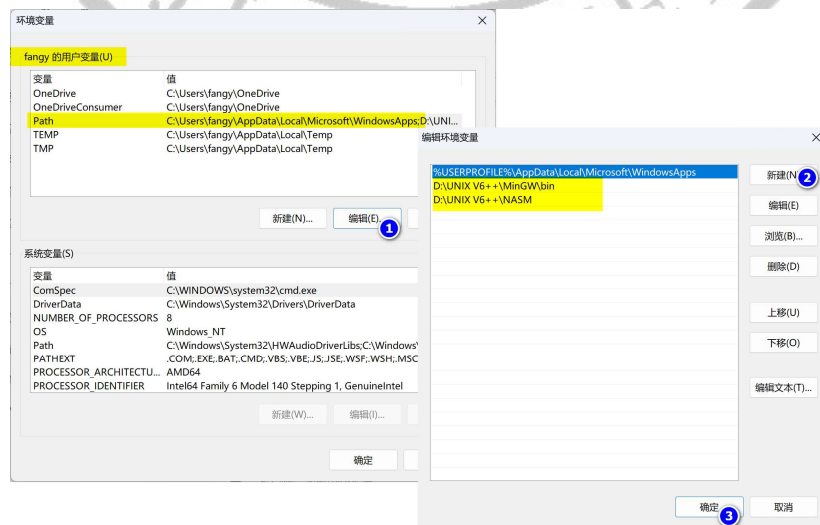


图 10：系统环境变量设置

5.4. ECLIPS 远程调试环境的配置

(1) 启动 eclipse

建议使用工具包中自带的这个版本, 如果自行下载和安装的其他版本, 有可能导致调试环境配置失败。需要注意的是, 安装 eclipse 之前, 请确保机器中已经安装好 java 虚拟机。如果没有安装, 可选择工具包中提供的 java 虚拟机进行安装。

(2) 将 oos 工程引入到 eclipse 中

通过 File 下的 import 菜单项, 将 oos 工程引入到 eclipse 中, 操作完成后, 可在 eclipse 中看到 oos 工程的全部源代码(如图 11 ~ 图 14 所示)。

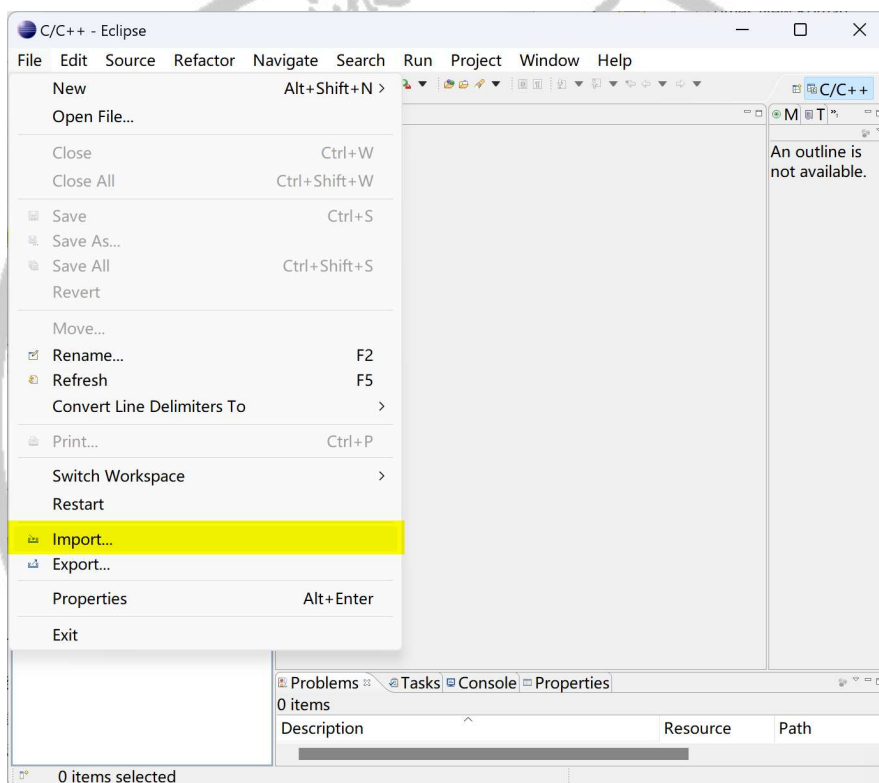


图 11

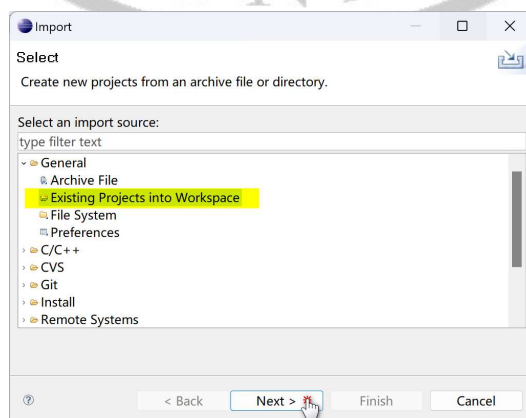


图 12

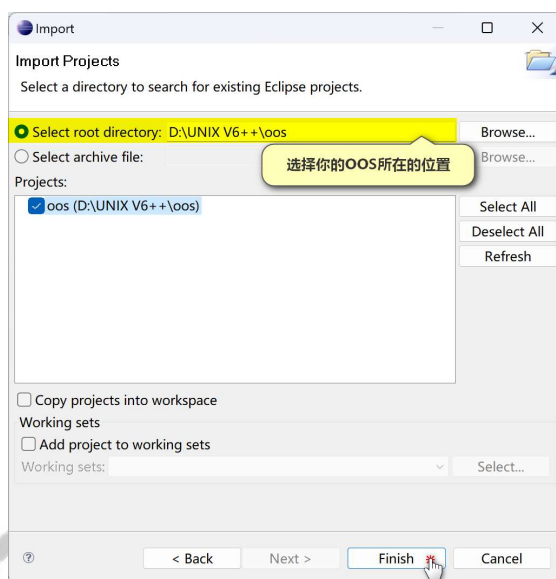


图 13

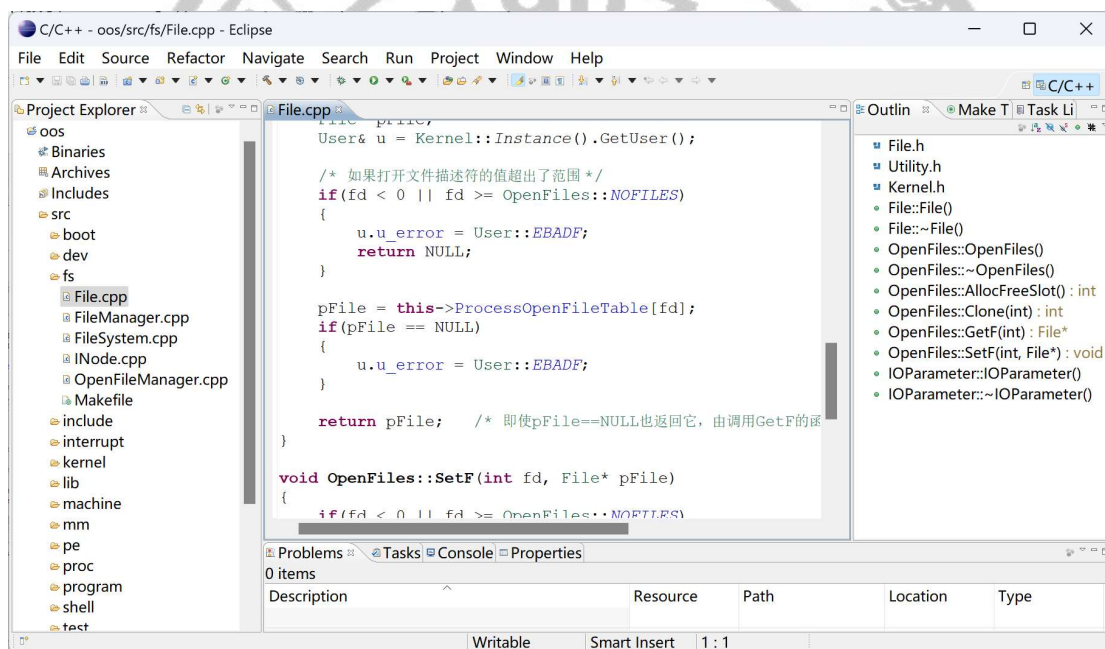


图 14

(3) 设置 oos 工程属性

选中工程名“oos”，进而选择 File 下的 Properties 菜单项（如图 15 所示），完成如下设置：（1）选择“C++ Build”，去除 ☐ Generate Makefiles automatically 选项，并设置 build directory 为 oos/src（如图 16 所示）；（2）在“C++ Build”下选择“tool Chain Editor”，其中的“Current tool Chain”选为“MinGW GCC”，“Current Builder”选为“Gnu Make Builder”，（如图 17 所示）。

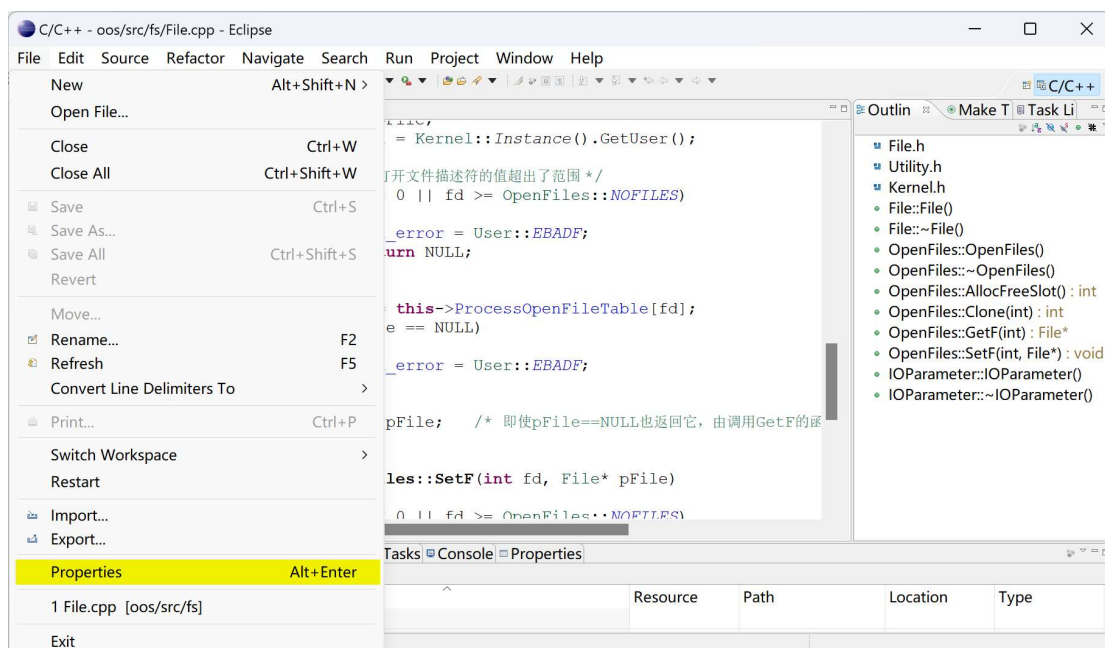


图 15

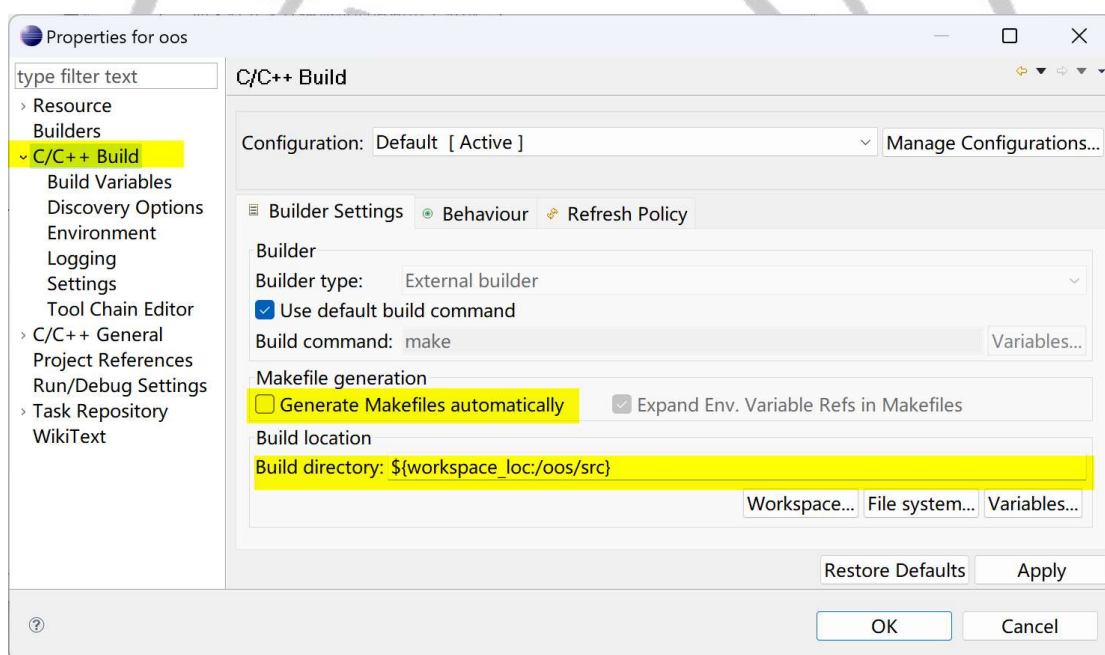


图 16

(4) 配置 gdb 远程调试

配置 gdb 远程调试的过程如下：

第一步：进入“Run”→“Debug Configurations”菜单项，选择“oos”，在“Main”选项卡中设置调试对象为 UNIX V6++ 的 kernel.exe 文件。点选 SelectOther，设置为 Standard Create Process Launcher 方式（图 18~21）。（若此处没有 oos，可选择 new 新建）

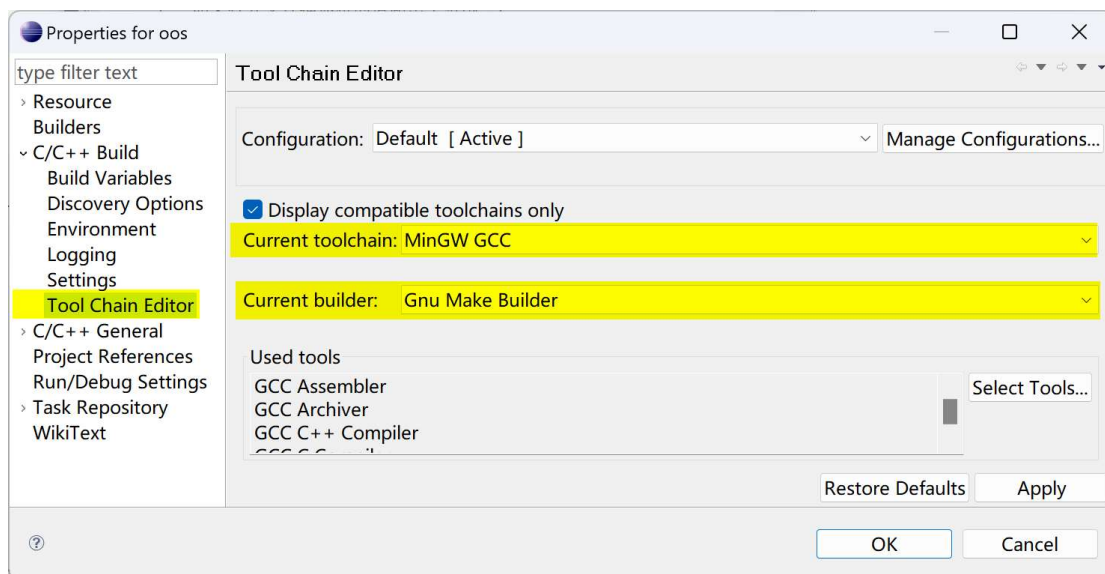


图 17

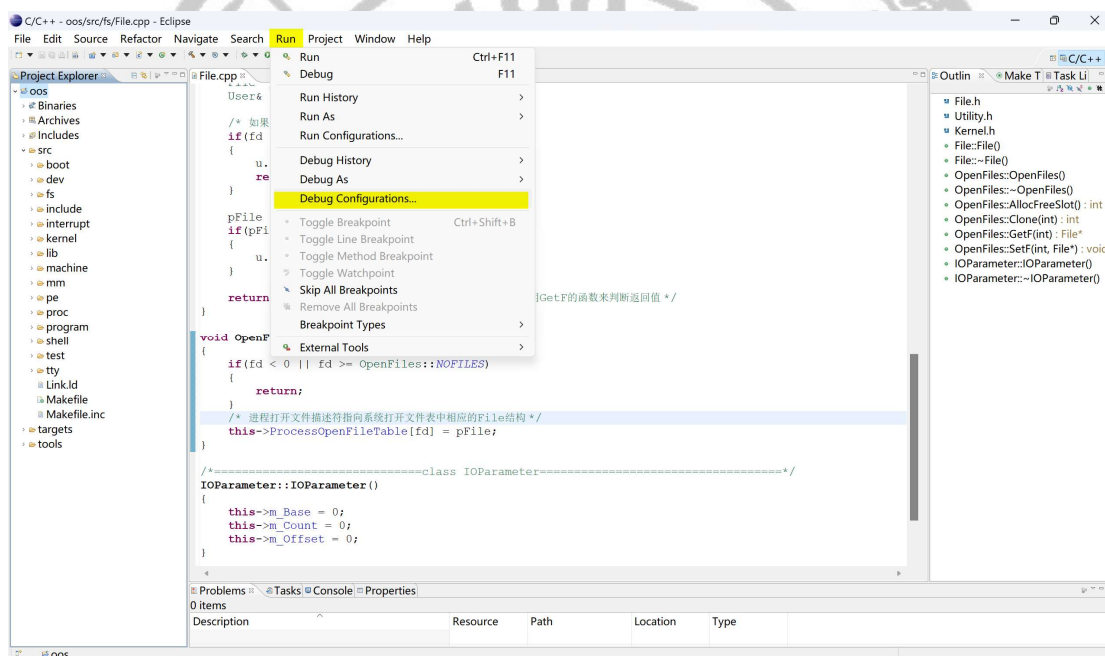


图 18

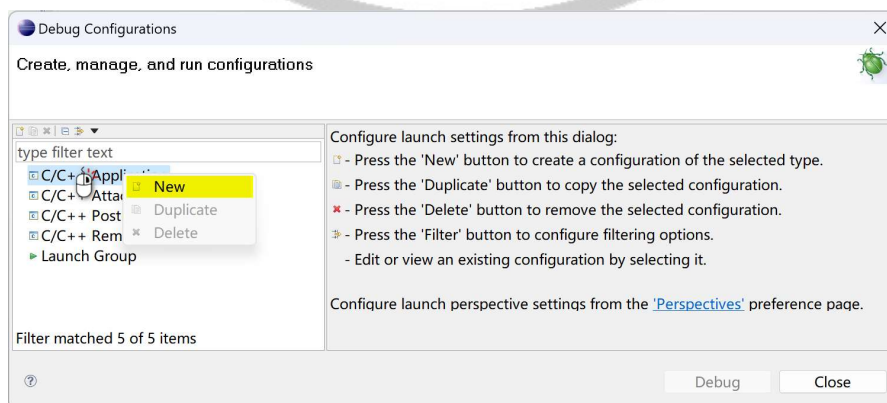


图 19

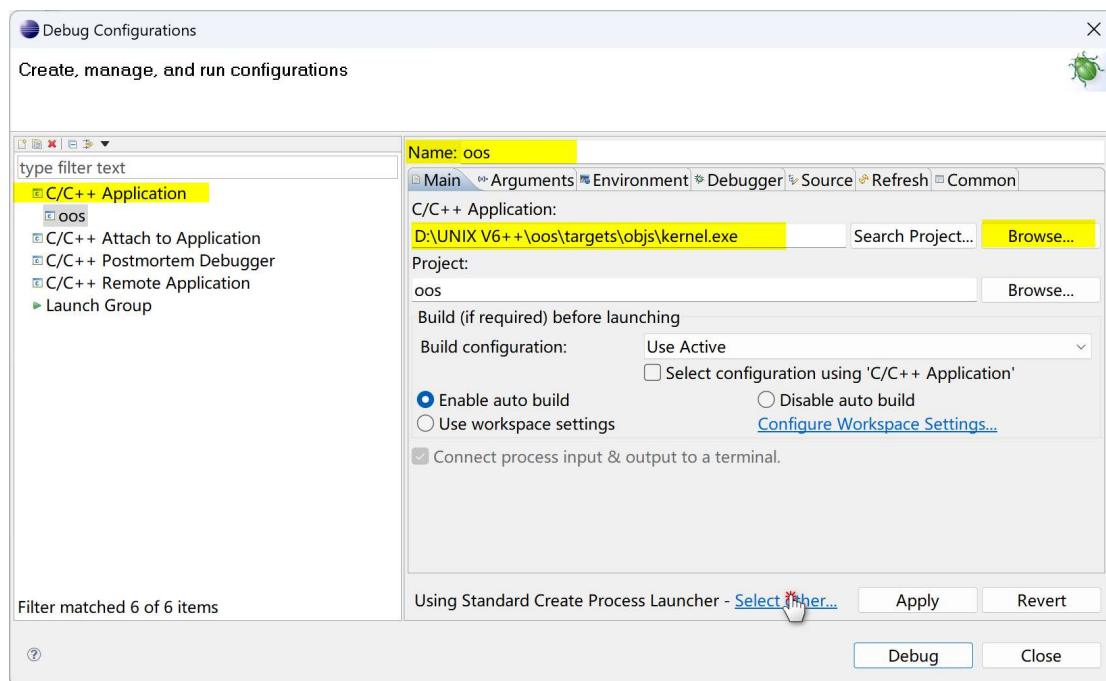


图 20

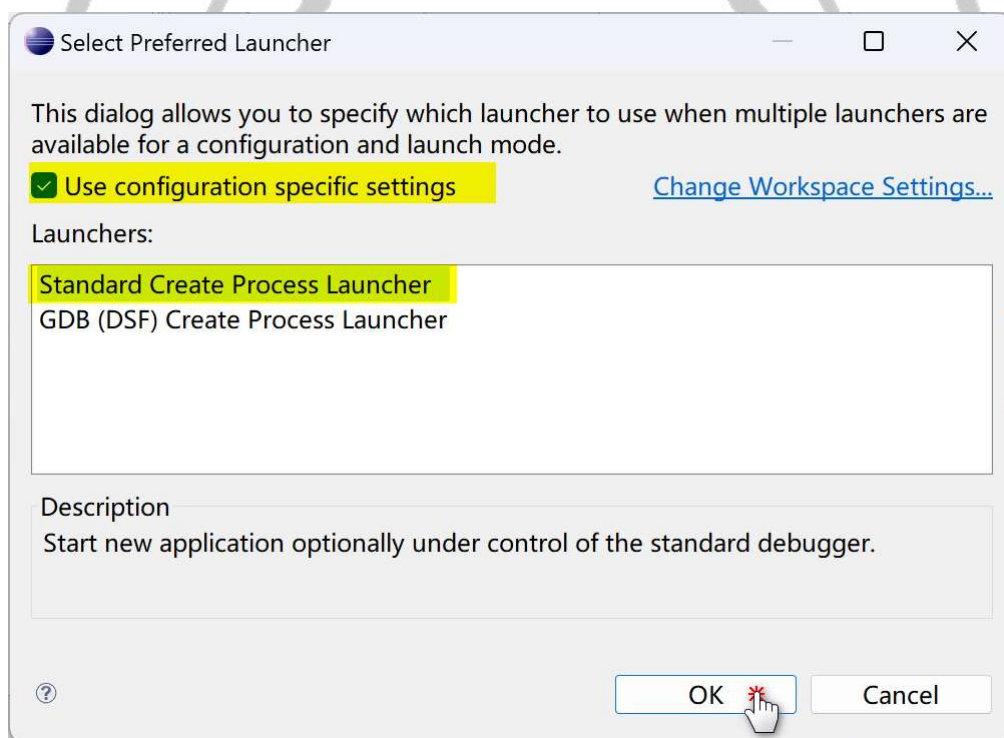


图 21

第二步：在“Debugger”选项卡中，设置调试器为“gdbserver”，起始调试点为“main0”函数，“connection”中设置连接方式为“TCP”，端口号为“1234”，与 bochs 虚拟机中的设置一致（如图 22）。

至此，我们完成了全部基于 eclipse 和 gdb 的远程调试环境的配置。

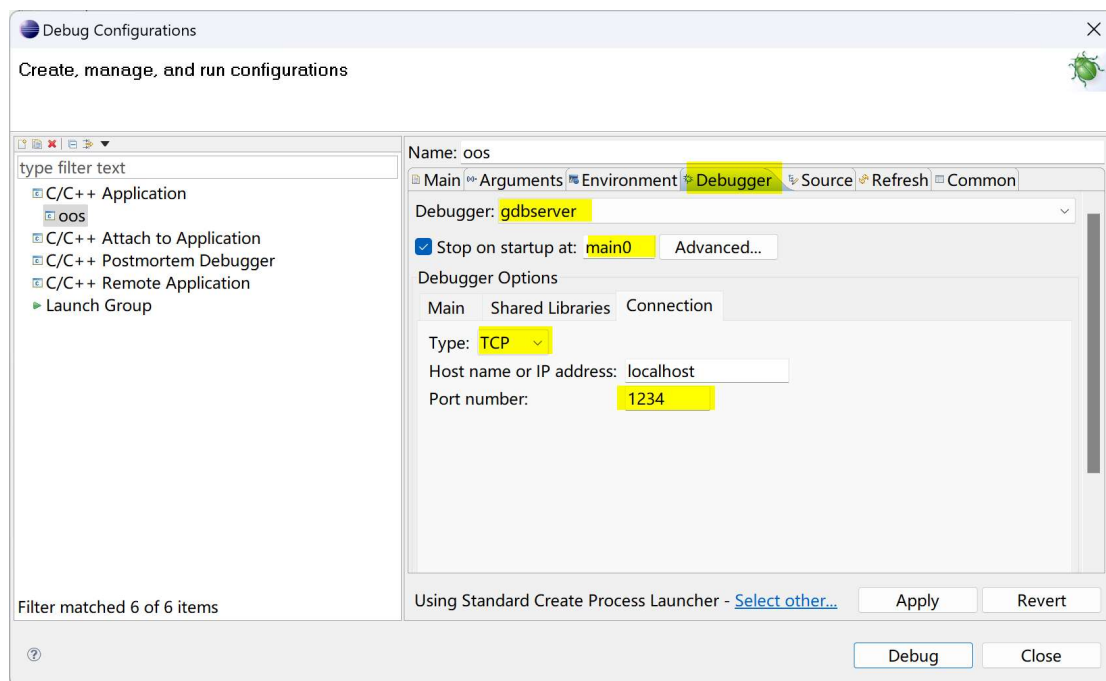



图 22

5.5. 开始调试

有了 eclipse 的远程调试环境之后，我们可以方便地在图形界面中完成对 UNIX V6++ 代码的调试工作。具体过程如下：

第一步：可以在源程序中的任意位置设置断点，见图 23 中 ①。

第二步：启动调试状态的 UNIX V6++。

第三步：在 eclipse 中开启调试，见图 23 中 ②，点击 。这时，会看到，UNIX V6++ 运行到了断点位置不再执行（如图 23）。此时，在 eclipse 中可以查看各个变量的值（如图 23 中 ③），或者通过“Registers”标签查看各个寄存器的值。

6. 实验报告要求

（1）（1 分）依照实验指导过程，完成 UNIX V6++ 的运行环境的安装与配置，执行几条简单的 UNIX 指令，关键步骤截图说明。

（2）（1 分）依照实验指导过程，完成 UNIX V6++ 的 eclipse 远程调试环境的安装与配置，关键步骤截图说明。

（3）（1 分）通过自己设置断点，观察 UNIX V6++ 的调试运行，查看变量与寄存器的值，截图说明。

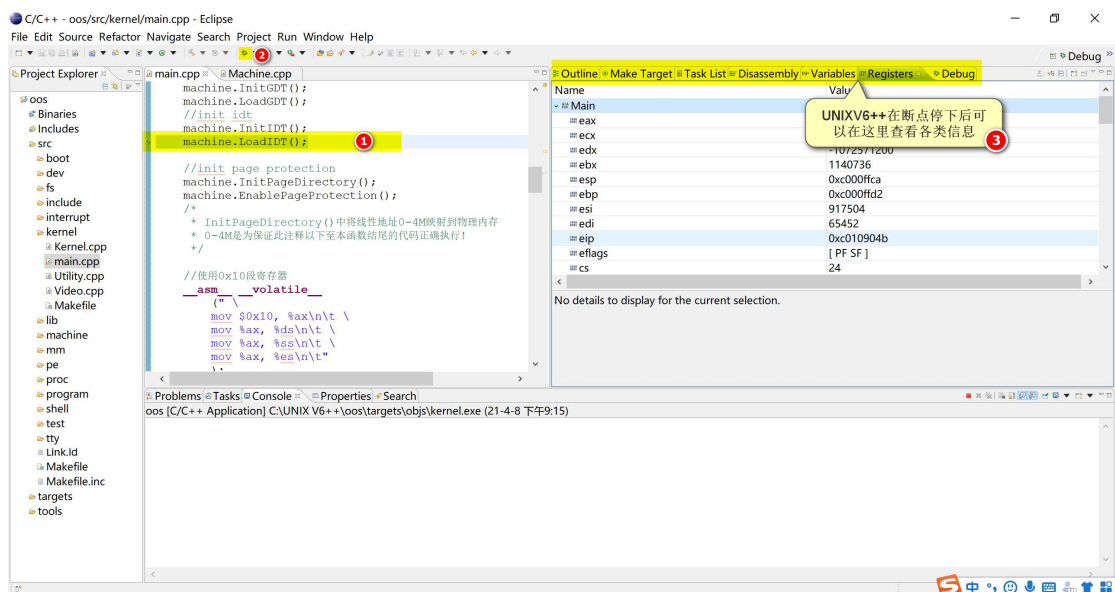


图 23：调试界面

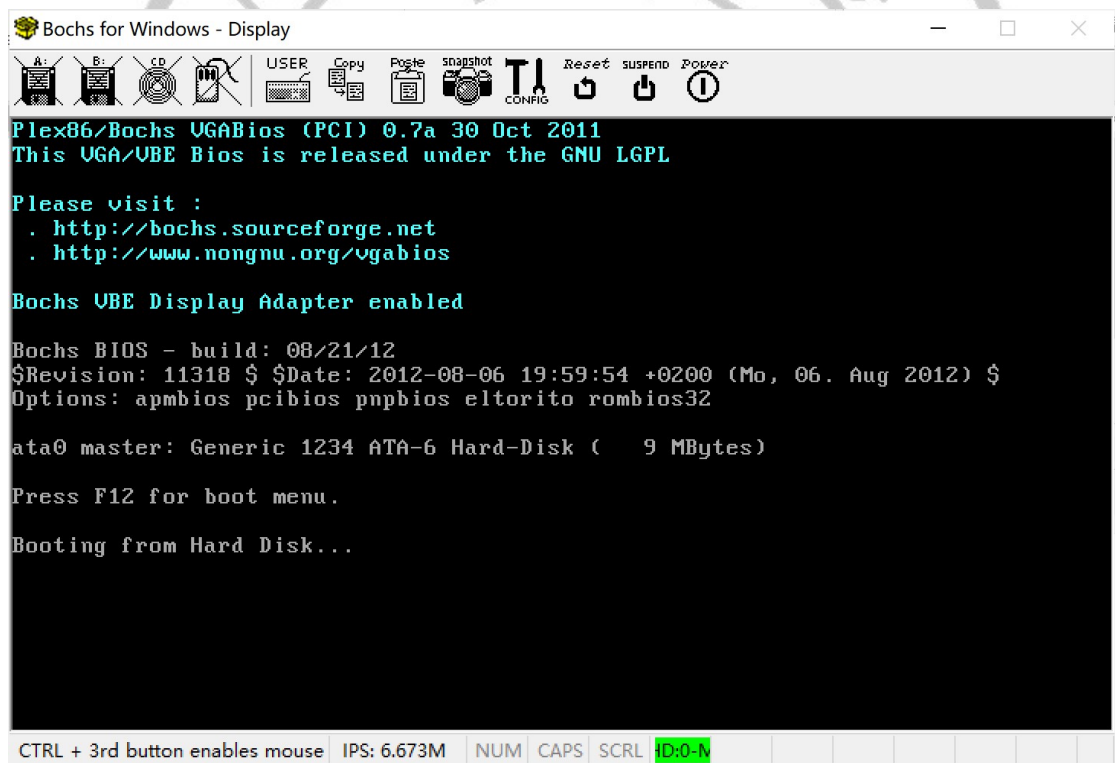


图 24：UNIXV6++停在断点

(4) (1 分) 在 eclipse 中观察整个 UNIX V6++源代码目录结构, 根据你的理解, 尝试给出每一个 oos/src/下的子目录中所包含文件的用途。