

**本科实验报告**

|  |  |
| --- | --- |
| 课程名称： | 计算机体系结构 |
| 姓 名： | 王俊 |
| 学 院： | 海洋学院 |
| 专 业： | 海洋工程与技术 |
| 学 号： | 3170100186 |
| 指导教师： | 翁恺 |

2020年 11 月 13 日

**Labs——发现跳转延时槽**

课程名称： 计算机体系结构 实验类型： 综合

实验项目名称： 发现跳转延时槽

学生姓名： 王俊 专业： 海洋工程与技术 学号： 3170100186

同组学生姓名： None 指导老师： 翁恺

实验地点： 曹光彪西-301 实验日期： 2020 年 11 月 13 日

1. 实验目的和要求

**Experiment Purpose:**

•Install the MIPS cross compiler

•write and adjust the C language program, and find the jump delay slot in the MIPS assembly code generated by the compilation.

**Experiment Apparatus:**

* Computer (Intel Core i5 or higher, 4GB RAM or higher) system
* Ubuntu 18.04
* Buildroot

**Experimental Materials：**

No

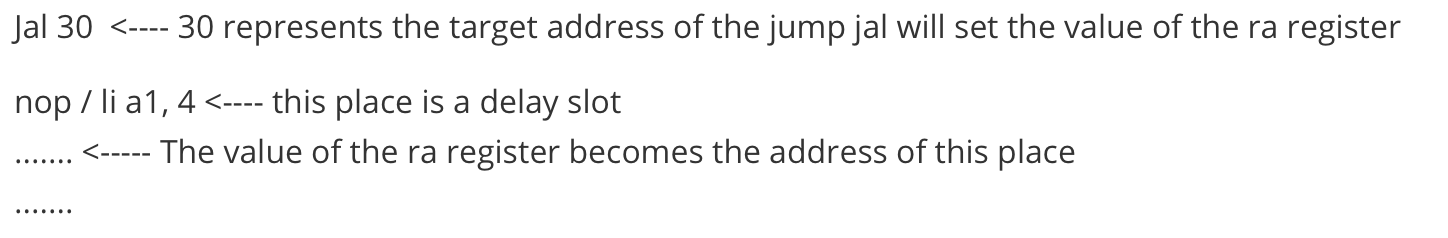
1. 实验内容和原理

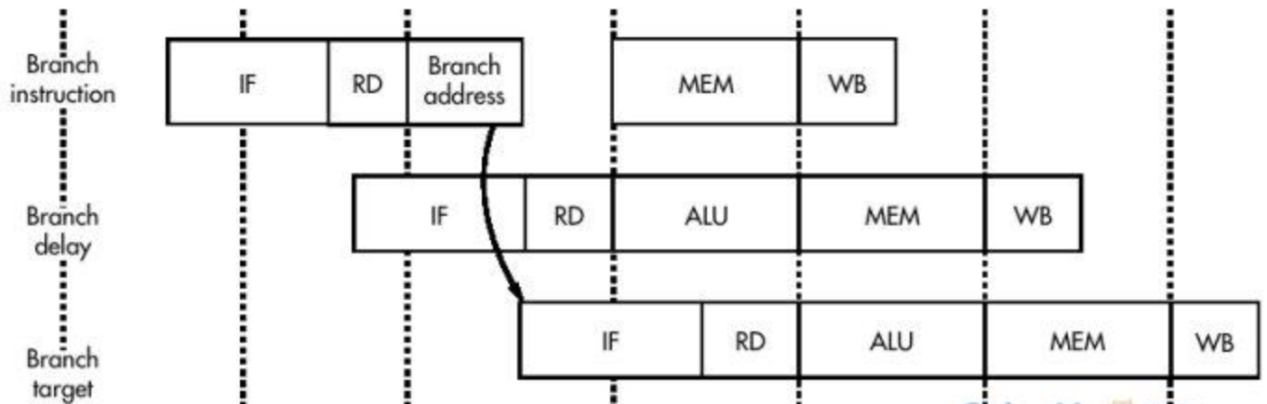
**2.1.mips delay slot**

**The main purpose of introducing a delay slot is to improve the efficiency of the pipeline, which is divided into the following two types:**

**1. Branch delay slot：**

Branch delay slot (Branch delay slot) is simply an instruction located after the branch instruction, which is always executed regardless of whether the branch occurs or not, and the following code shows that the instruction located in the branch delay slot precedes the target The instruction is submitted (commit means execution).

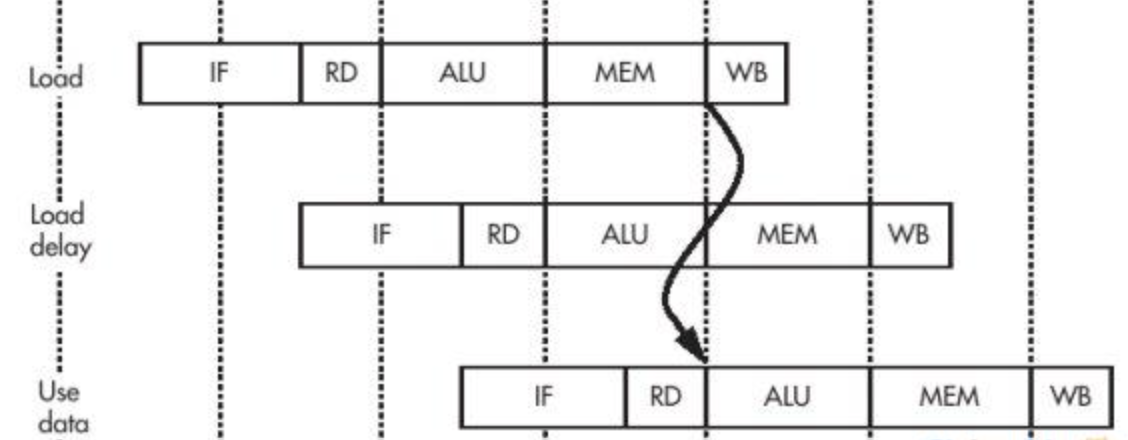




**2. Storage delay slot:**

The data of the data load instruction can only be obtained from the cache or memory after the ALU stage of the next instruction is started, so the next instruction cannot use the data. If the next instruction compulsorily reference the data, then the instruction will stop running (in the ALU Stage), wait for the data load instruction to complete before starting to run.

As shown below:



**2.2.Delay slot exception return address is the address of the previous instruction:**

The general CPU branch jump instruction flow is: branch jump instruction -> target jump address instruction.

But the branch jump instruction flow of MIPS is: branch jump instruction -> delay slot instruction -> instruction of the target jump address, and the delay slot instruction is inserted in the middle operation.

If the PC returns to the delay slot instruction address when the interrupt returns after the delay slot address is interrupted, the re-executed instruction flow is: delay slot instruction -> (delay slot instruction address + 4) address instruction, no jump Up!

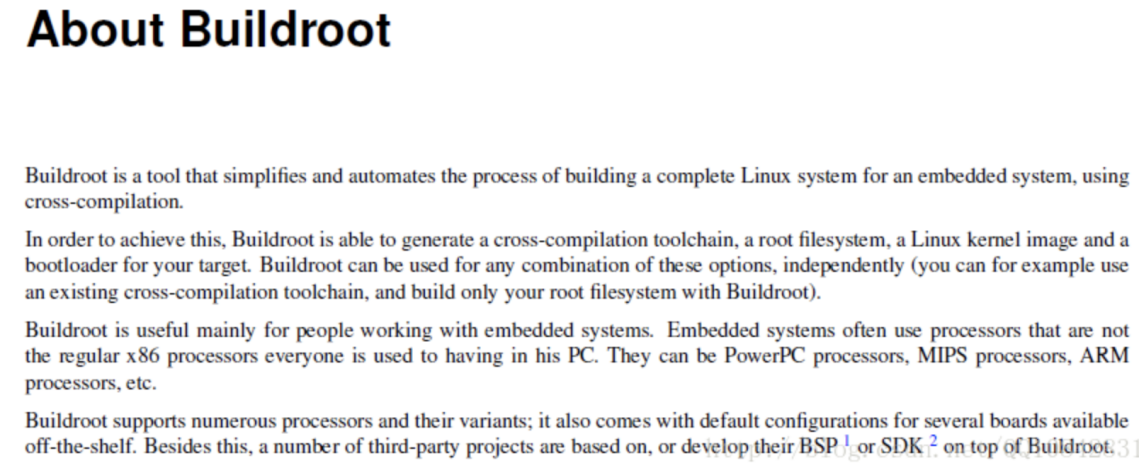
This is not the interrupted instruction flow at all. In order to restore the original instruction flow, the jump instruction before the delay slot needs to be reloaded into the pipeline (the instruction in the delay slot will be executed again, but it is not necessarily an exception) .

So the address returned after the delay slot interrupt is the address of the previous jump instruction.

1. 实验过程

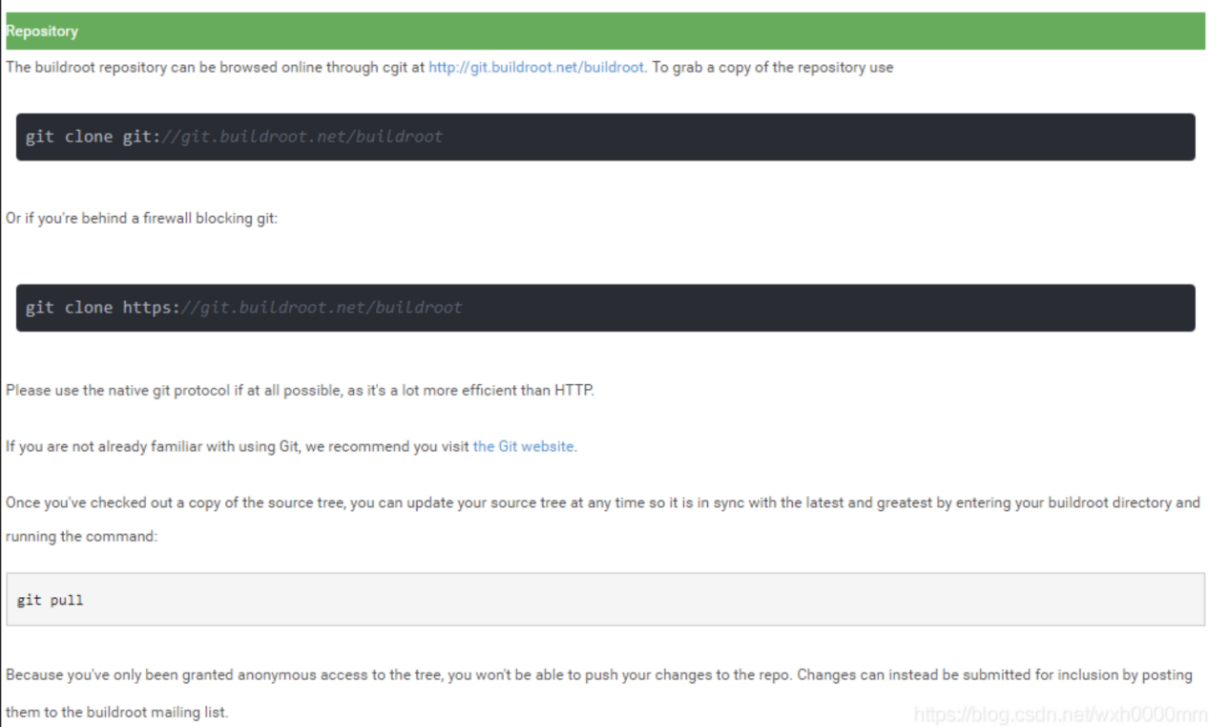
**3.1.MIPS交叉编译环境的搭建（Buildroot）**

* **what is Buildroot ?**

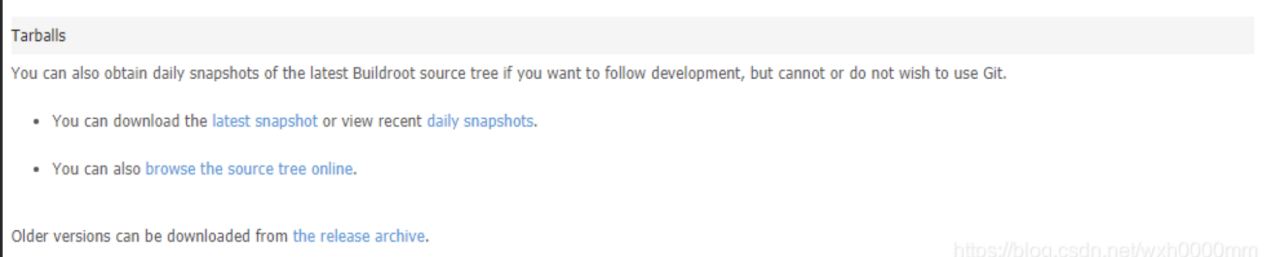


* **Buildroot工程源码的下载，官方提供了两种方法。**

1.通过github的方式获取：



1. 直接下载的方式

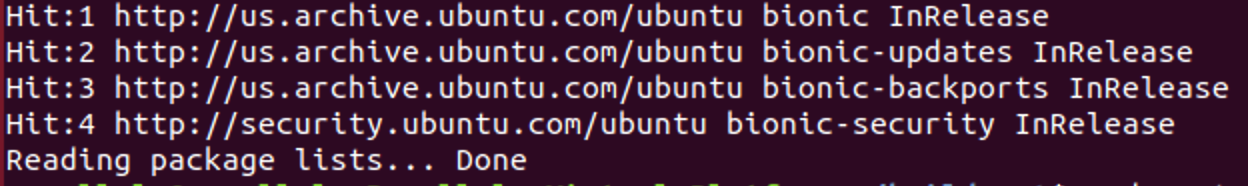


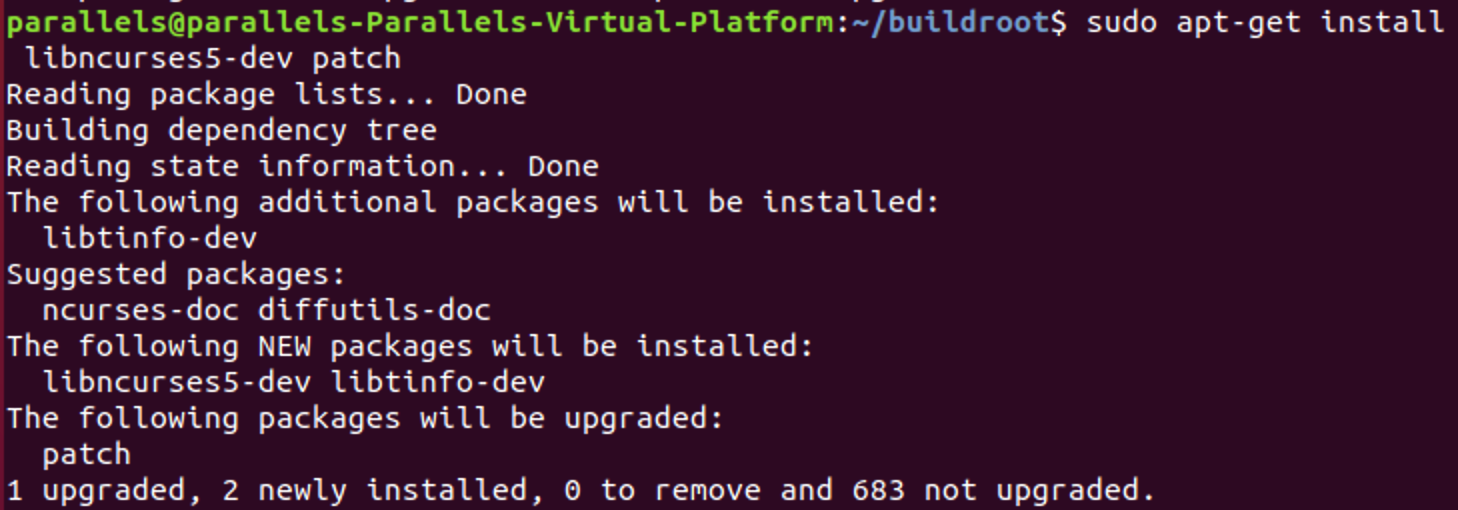
* **这里我选择了第二种：**



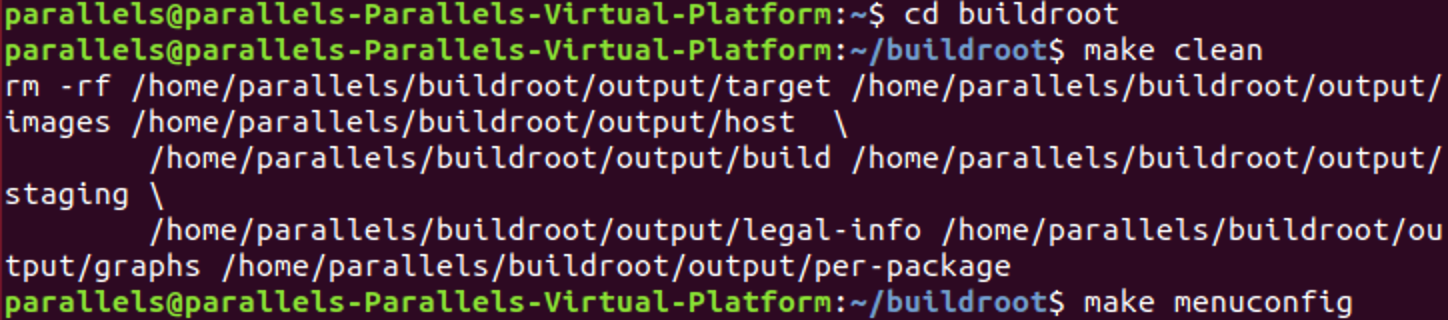
安装依赖库：

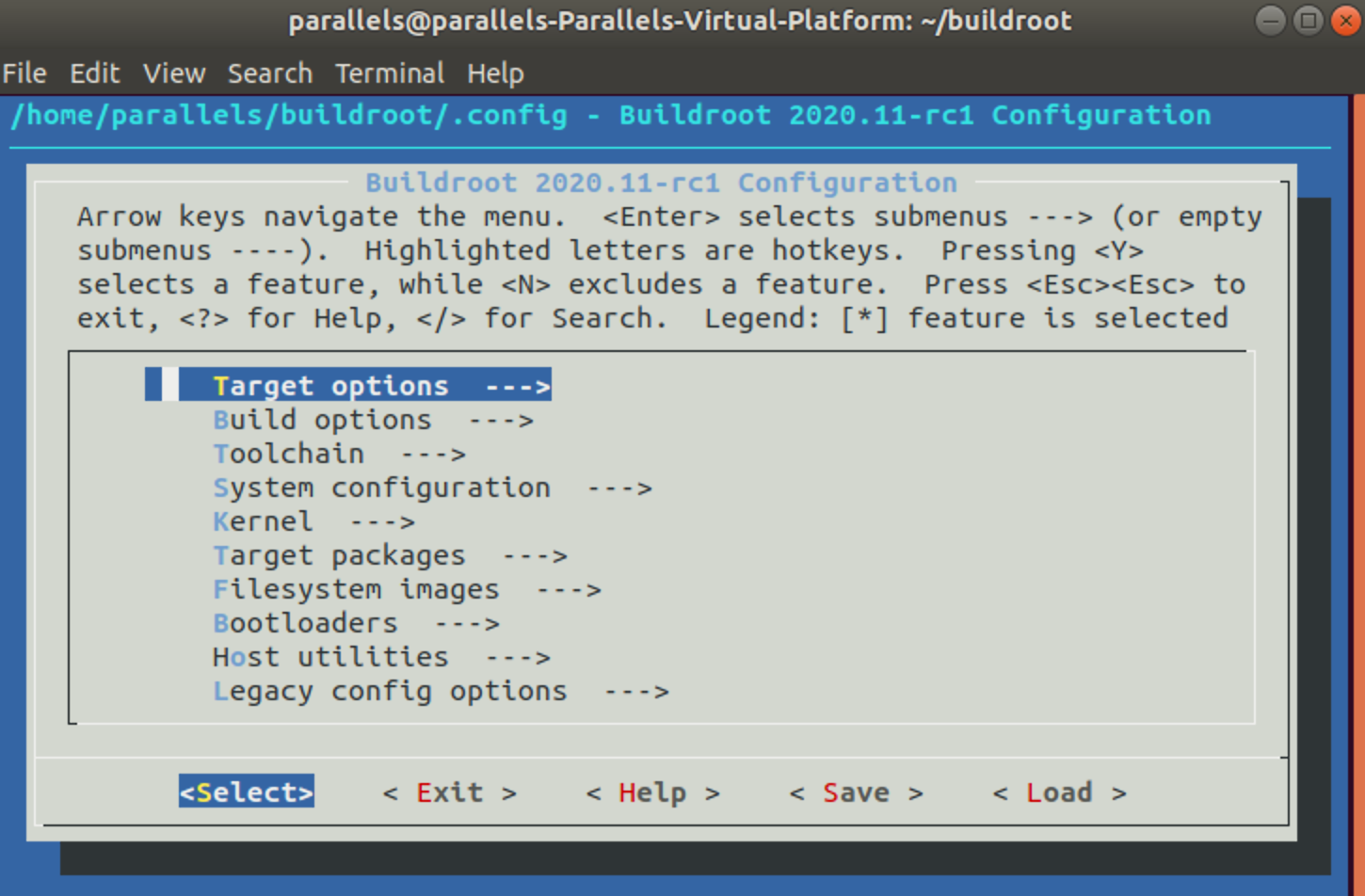






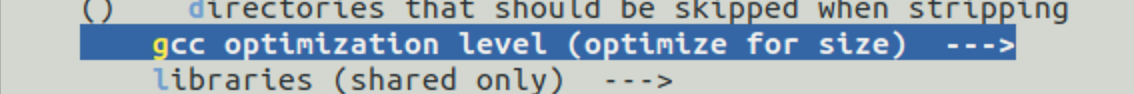
然后下载解压文件，进行编译：





* **将优化等级设置成level 2**

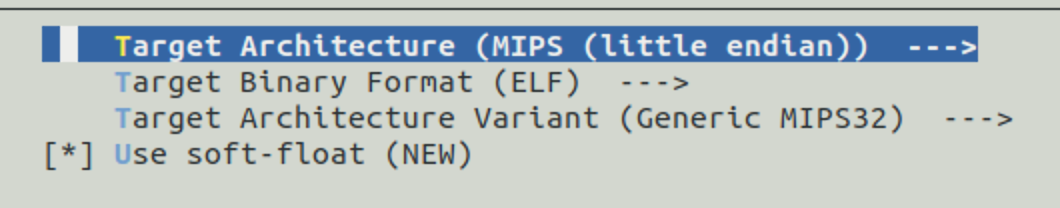
**原来：**



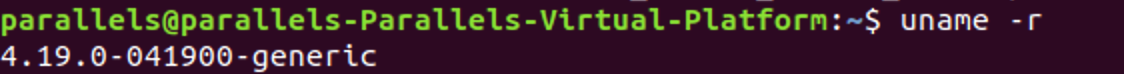
设置后：



* **设置mips语言**

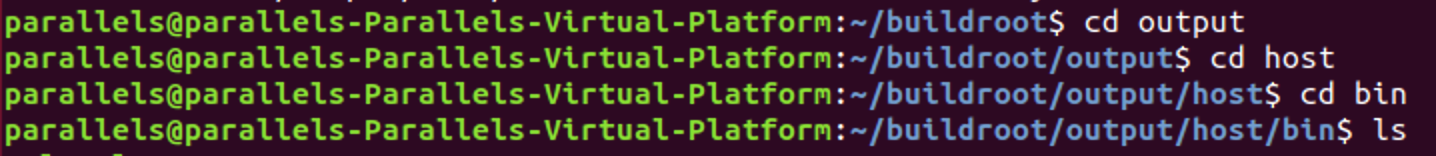


* **查看kernel，并设置：**

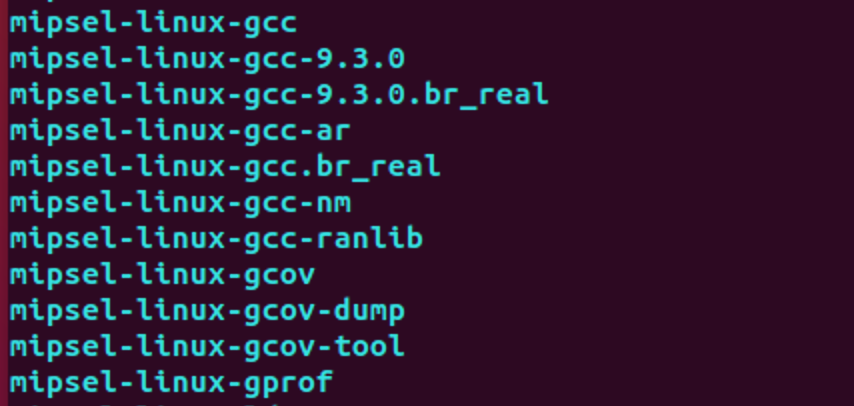


然后输入指令make，并等待：

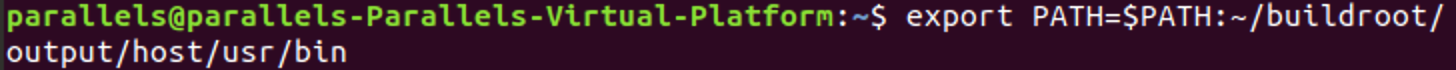
***经过一段时间的编译完成以后，在Buildroot的根目录下会增加一个output文件，其中包含已经编译好的文件。可以在 buildroot/output/host/usr/bin目录下找到生成的交叉编译工具，mips的编译器就是该目录下的mipsel-linux-gcc。***



并找到目录下的mipsel-linux-gcc

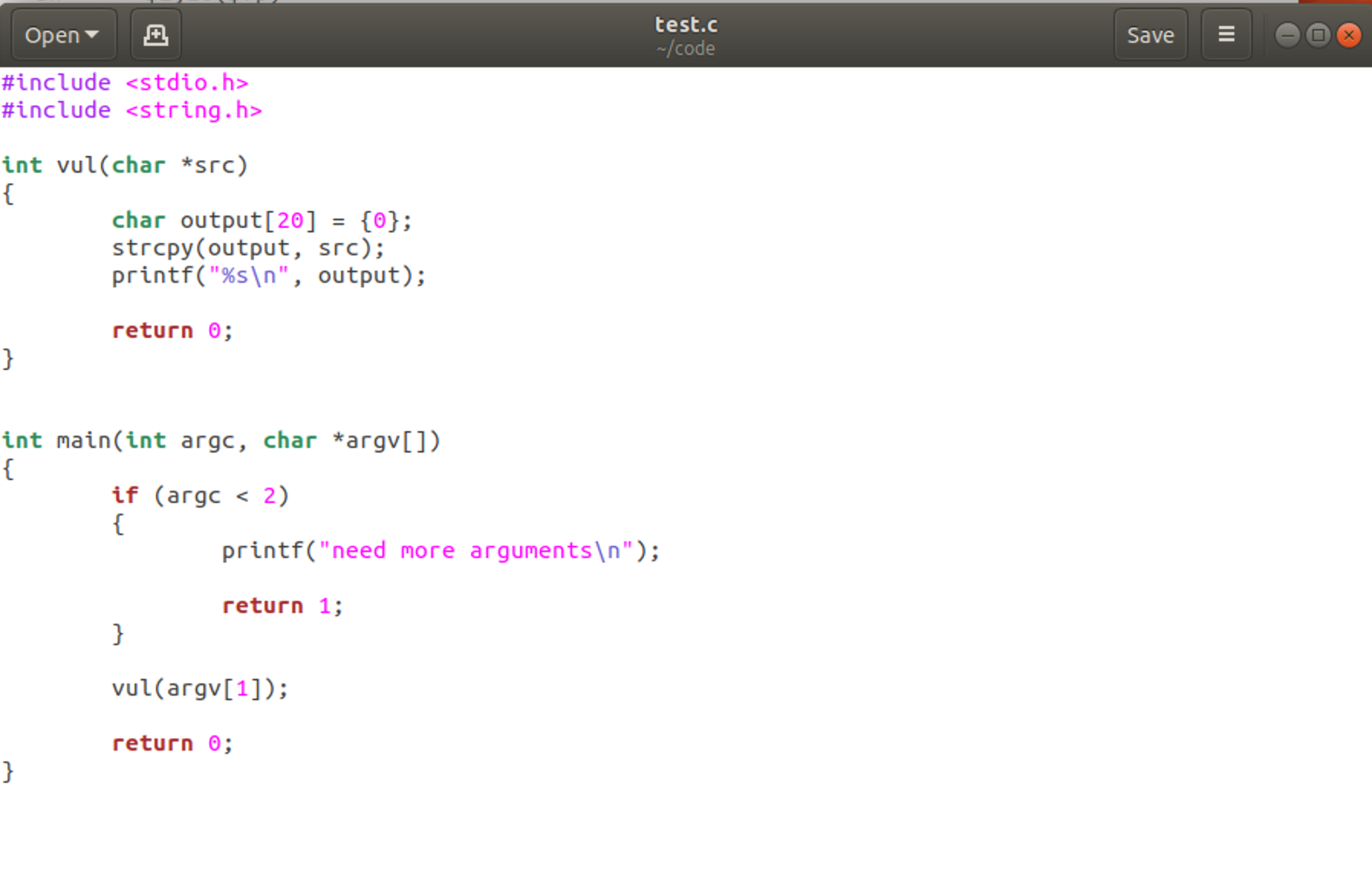


创建环境变量,然后我们就可以使用了



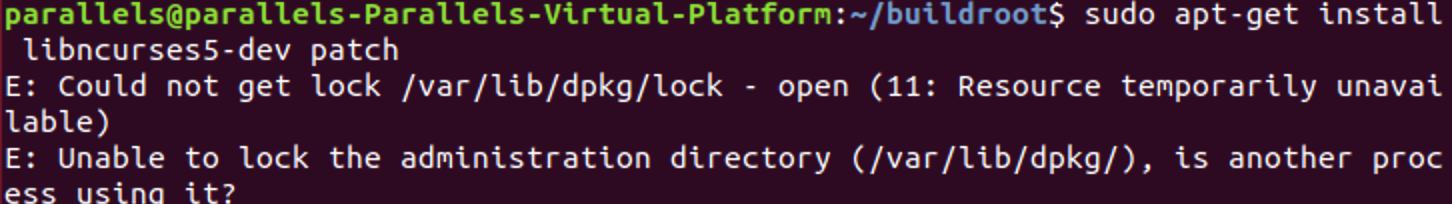
1. 实验结果分析

* Test.c：



五、实验总结与反思

**1.在下载libcurses5-dev patch的时候出现了问题**



问题原因：主要是因为apt还在运行。

解决方案：

1：查找所有apt相关的进程，并用命令杀死。



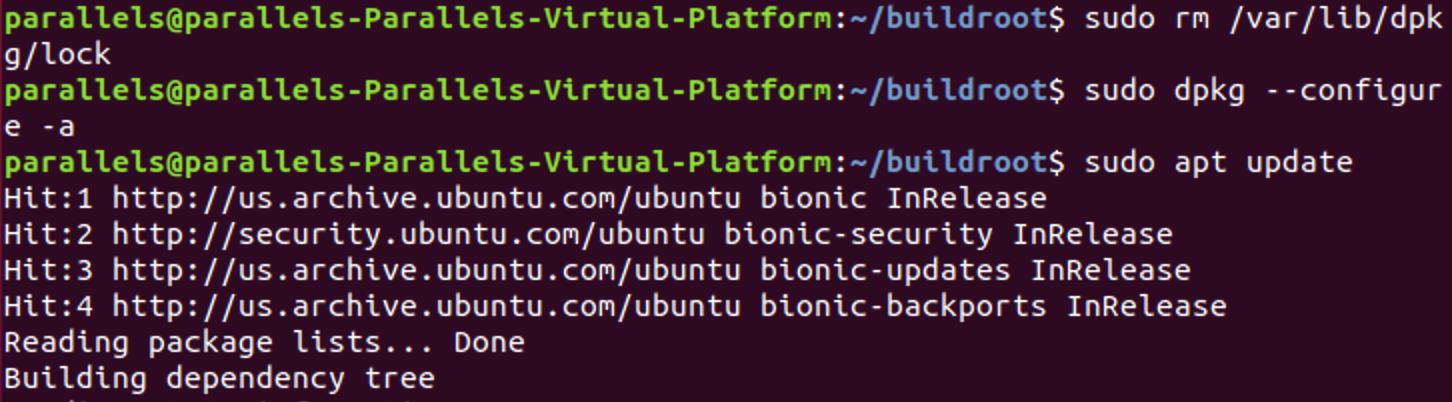
2：删除锁定文件

锁定的文件会阻止 Linux 系统中某些文件或者数据的访问，这个概念也存在于 Windows 或者其他的操作系统中。

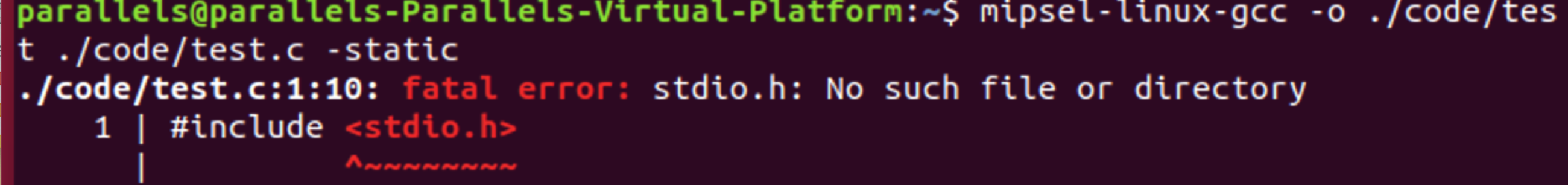
一旦你运行了 apt-get 或者 apt 命令，锁定文件将会创建于 /var/lib/apt/lists/、/var/lib/dpkg/、/var/cache/apt/archives/ 中。

这有助于运行中的 apt-get 或者 apt 进程能够避免被其它需要使用相同文件的用户或者系统进程所打断。当该进程执行完毕后，锁定文件将会删除。

所以：1：移除对应目录下的锁文件：2：强制重新配置软件包：3：更新软件包源文件



**2.在编译的时候出现了问题**

找不到头文件，于是直接使用第三种方法进行下载：

其实MIPS程序的交叉编译工具gcc不必非得自己编译得到，网站https://uclibc.org/上已经有编译后的mips程序的交叉编译工具提供给我们下载，在ubuntu系统上，可以使用脚本 config.sh进行下载，脚本文件的内容如下：

**Config.sh:**

