



# 止戈

(<http://www.wjgbaby.com/>)

## Linux 0.11实验一： 环境配置

2018年4月30日 (<http://www.wjgbaby.com/2018/04/30/linux0-11%E5%AE%9E%E9%AA%8C%E4%B8%80%E7%BC%9A%E7%8E%AF%E5%A2%83%E9%85%8D%E7%BD%AE/>) / 止戈 (<http://www.wjgbaby.com/author/root/>) / %1 \$ S

# 主要平台和工具简介

## x86模拟器Bochs

Bochs是一个免费且开放源代码的IA-32(x86)架构PC机模拟器。在它模拟出的环境中可以运行Linux、DOS和各种版本的Windows等多种操作系统。而Bochs本身具有很高的移植性，可以运行在多种软硬件平台之上，这也是我们选择它做为本书的指定模拟器的主要原因。如果您想拥抱自由的Linux，那么Bochs几乎是您的不二选择。如果您想继续把自己绑定在Windows平台上，那么除了Bochs，您还可以选用VMware或者Microsoft Virtual PC。它们是最著名虚拟机软件，而且都可以免费使用。因为Bochs的是模拟器，其原理决定了它的运行效率会低于虚拟机。但对于本书所设计的实验来说，效率上的差别很不明显。而且，Bochs有虚拟机无可比拟的调试操作系统的能力，所以我们更建议您选用Bochs。hit-oslab已经内置了bochs，本书后文假定的缺省环境也是Bochs。

关于Bochs的更详细的介绍请访问它的主页及Bochs使用手册

(<https://hoverwinter.gitbooks.io/hit-oslab-manual/content/bochs-manual.html>)。

## GCC编译器

GCC是和Linux一起成长起来的编译器。Linux最初的版本就是由GCC编译的。现在GCC也是在自由软件领域应用最广泛的编译器。所以，我们也选择GCC做为本书实验的指定编译器。

## DB调试器

GDB调试器是GCC编译器的兄弟。做为自由软件领域几乎是唯一的调试器，它秉承了\*nix类操作系统的一贯风格，采用纯命令行操作，有点儿类似dos下的debug。关于它的使用方法请看GDB使用手册。

## Ubuntu (GNU/Linux)

Ubuntu也许不是目前最好用的Linux桌面发行版，但它一定是最流行的。主要特点是易用，非常的易用。

现在，已经有越来越多的人开始用Ubuntu完全代替Windows，享受更加自由、安全、守法的感觉。Ubuntu的主页是<http://www.ubuntu.com/> (<http://www.ubuntu.com/>)，这里不仅可以免费下载到iso文件，甚至能免费申领Ubuntu的安装光盘。

我们强烈建议您在Ubuntu下做实验。因为有些实验内容涉及到在自己改进的Linux 0.11下，运行自己编的应用程序。被改进的功能都是高版本Linux内核已经具有的，在其上确认自己编写的应用程序无误后，再用之测试自己改进的Linux 0.11，可以更有信心些。

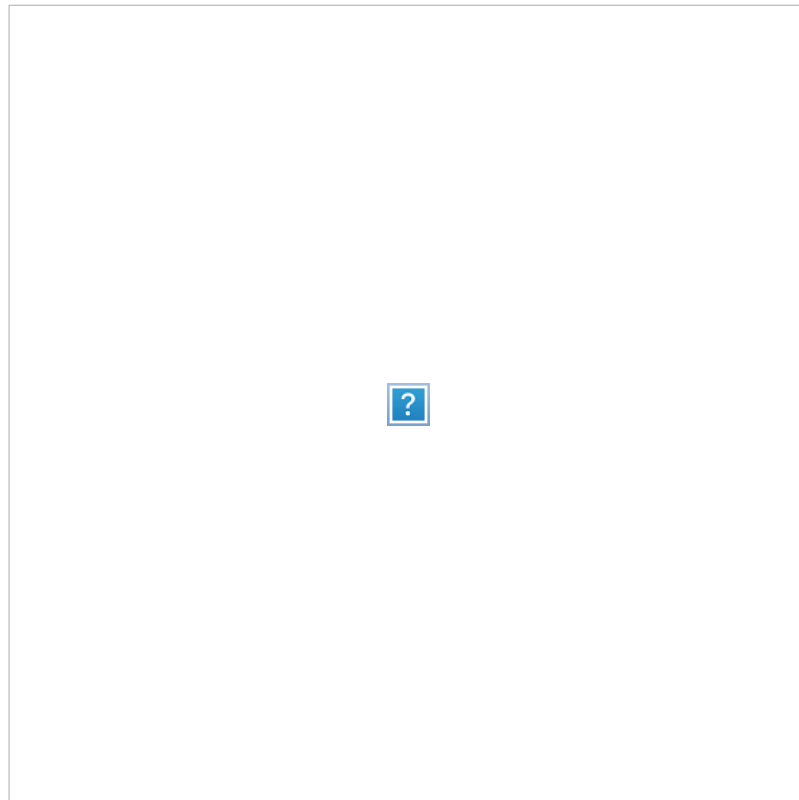
# 安装指南

安装指南链接，这篇文章比较好，它把十几条配置环境的命令行都整理好集成到了一条命令中，对于新人来说可以不用太在意那些命令行：

<https://github.com/DeathKing/hit-oslab> (<https://github.com/DeathKing/hit-oslab>)

setup.sh具体操作细节，有兴趣可以去看看：

<https://github.com/Wangzhike/HIT-Linux-0.11/blob/master/0-prepEnv/%E5%87%86%E5%A4%87%E5%AE%89%E8%A3%85%E7%8E%AF%E5%A2%83.md> (<https://github.com/Wangzhike/HIT-Linux-0.11/blob/master/0-prepEnv/%E5%87%86%E5%A4%87%E5%AE%89%E8%A3%85%E7%8E%AF%E5%A2%83.md>)



(<http://wu.wjgbaby.com/wp-content/uploads/2018/04/18043001.png>)

## 使用方法

### 准备活动

```
1 $ cd ~/oslab
2
```

把当前目录切换到oslab下，用pwd命令确认，用“ls -l”列目录内容。本实验的所有内容都在本目录或其下级目录内完成。

### 编译内核

“编译内核”比“编写内核”要简单得多。首先要进入linux-0.11目录，然后执行：

```
1 $ make all
2
```

因为“all”是最常用的参数，所以可以省略，只用“make”，效果一样。

在多处理器的系统上，可以用-j参数进行并行编译，加快速度。例如双CPU的系统可以：

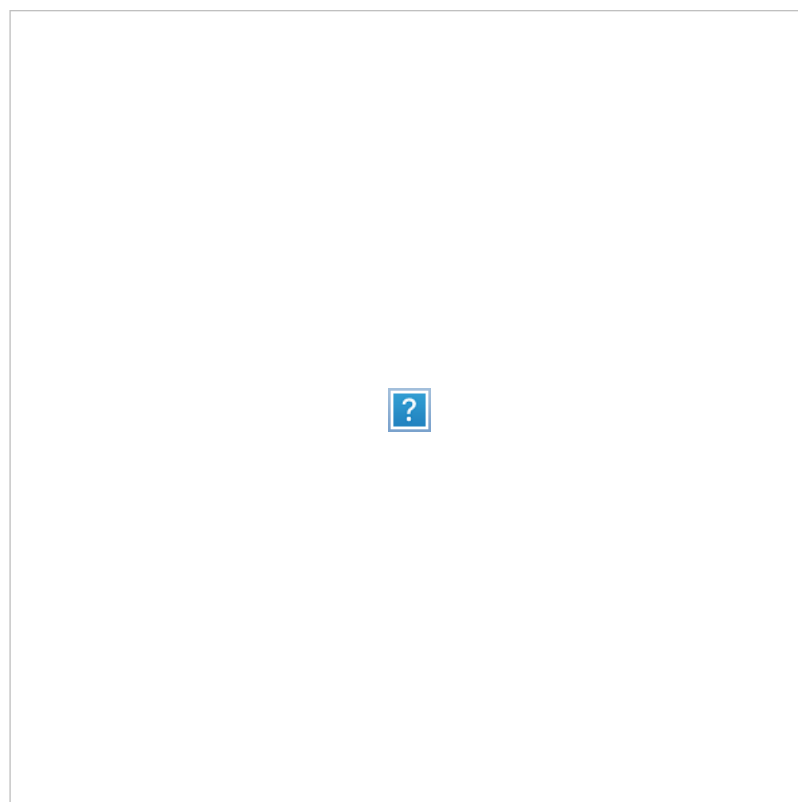
```
1 $ make -j 2
2
```

make命令会显示很多很多很多的信息，你可以尽量去看懂，也可以装作没看见。只要最后几行中没有“error”就说明编译成功。最后生成的目标文件是一个软盘镜像文件——linux-0.11/Image。如果将此镜像文件写到一张1.44MB的软盘上，就可以启动一台真正的计算机。

linux-0.11目录下是全部的源代码，很多实验内容都是要靠修改这些代码来完成。修改后需要重新编译内核，还是执行命令：

```
1 $ make all
2
```

make命令会自动跳过未被修改的文件，链接时直接使用上次编译生成的目标文件，从而节约编译时间。但如果重新编译后，你的修改貌似没有生效，可以试试先“make clean”，再“make all”。“make clean”是删除上一次编译生成的所有中间文件和目标文件，确保是在全新的状态下编译整个工程。



(<http://wu.wjgbaby.com/wp-content/uploads/2018/04/18043002.jpg>)

## 运行和调试

在Bochs中运行最新编译好的内核很简单，在oslab目录下执行：

```
1 $ ./run
2
```

如果出现Bochs的窗口，里面显示linux的引导过程，最后停止在“[/usr/root/]#”，表示运行成功。

内核调试分为两种模式：汇编级调试和C语言级调试。

汇编级调试需要执行命令：

```
1 $ ./dbg-asm
2
```

可以用命令help来查看调试系统用的基本命令。更详细的信息请查阅Bochs使用手册。

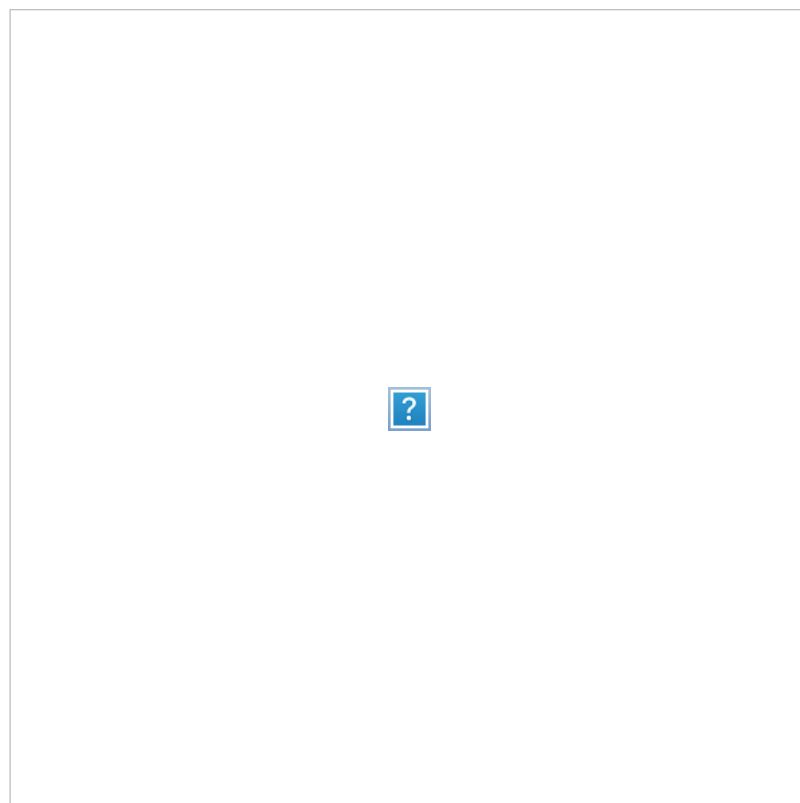
C语言级调试稍微复杂一些。首先执行如下命令：

```
1 $ ./dbg-c
2
```

然后再打开一个终端窗口，进入oslab目录后，执行：

```
1 $ ./rungdb
2
```

新终端窗口中运行的是GDB调试器。关于gdb调试器请查阅GDB使用手册。



(<http://wu.wjgbaby.com/wp-content/uploads/2018/04/18043003.jpg>)

参考链接：

操作系统原理与实践 (<https://www.shiyanlou.com/courses/115>)

课程链接：

操作系统之基础 (<http://mooc.study.163.com/course/1000002004?tid=2001426000#/info>)

操作系统之进程与线程 (<http://mooc.study.163.com/course/1000002008?tid=2001427001#/info>)

操作系统之内存管理 (<http://mooc.study.163.com/course/1000003007?tid=2001427002>)

操作系统之外设与文件系统 (<http://mooc.study.163.com/course/1000002009?tid=2001426003#/info>)

所需资源下载:

GitHub (<https://github.com/hoverwinter/HIT-OSLab/tree/master/Resources>)



标签: 操作系统

(<http://www.wjgbaby.com/tag/%e6%93%8d%e4%bd%9c%e7%b3%bb%e7%bb%9f/>)

光照系统之灯光探头组

(<http://www.wjgbaby.com/2018/04/27/%e5%85%89%e7%85%a7%e7%b3%bb%e7%bb%9f%e4%b9%8b%e7%81%af%e5%85%89%e6%8e%a2%e5%a4%b4%e7%bb%84/>)

CS: 数值和编码

(<http://www.wjgbaby.com/2018/05/05/%e8%ae%a1%e7%ae%97%e6%9c%ba%e7%b3%bb%e7%bb%9f%ef%bc%9a%e6%95%b0%e5%80%bc%e5%92%8c%e7%b9%96%e7%a0%81/>)

(<http://www.dddcode.cn/>) 木尘痕 (<http://www.muchenhcn.com/>) 腾讯云优惠券  
(<https://www.laomulu.com/>) 技术人生 (<http://www.luzexi.com/>)