1. 准备源码

从 Bochs x86 PC emulator - Browse /bochs/2.7 at SourceForge.net 下载源码 `bochs-2.7.tar.gz` ，解压到指定目录。

```
$ cd ~/workspace
$ tar -xvf bochs-2.7.tar.gz
$ cd bochs-2.7
```

2. 编译

首先进行环境准备，本次编译使用 SDL2 库用作图形显示。

```
$ sudo apt install build-essential libsdl2-dev
```

修改配置文件，bochs默认准备了一些配置脚本可以快速配置，本次就以 `.conf.linux` 配置脚本为基础，然后修改如下：

```
......

plugins)
    ########################################################################
    # configuration 2 for release binary RPMs
    # Include plugins, every possible gui.
    ########################################################################
    ./configure  \
                --enable-all-optimizations \
                --enable-cpu-level=6 \
                --enable-pci \
                --enable-clgd54xx \
                --enable-voodoo \
                --enable-usb \
                --enable-usb-ohci \
                --enable-busmouse \
                --enable-plugins \
                --with-all-libs \
                --with-sdl2 \
```

```
                --prefix=~/bin/bochs \          # 如果~不识别，请改为/home/用户名
                ${CONFIGURE_ARGS}
    ;;
......
```

运行配置脚本

```
$ sh .conf.linux
$ make -j4     # 编译
$ make install   # 安装
```

3. 准备配置文件和镜像

新建配置文件，也可以使用 `.bochsrc` 为基础，自行修改。

```
$ cat bochsrc.bxrc

# configuration file generated by Bochs
#plugin_ctrl: voodoo=true, unmapped=true, biosdev=true, speaker=true, extfpuirq=t
plugin_ctrl: voodoo=true, unmapped=true, biosdev=true, speaker=true, extfpuirq=tr
#config_interface: win32config
#display_library: win32
# memory: host=32, guest=32
romimage: file="/home/hupeng/bin/bochs/share/bochs/BIOS-bochs-latest", address=0x
vgaromimage: file="/home/hupeng/bin/bochs/share/bochs/VGABIOS-lgpl-latest"
# boot: floppy
floppy_bootsig_check: disabled=0
# floppya: type=1_44
# no floppyb
ata0: enabled=true, ioaddr1=0x1f0, ioaddr2=0x3f0, irq=14
# ata0-master: type=none
ata0-slave: type=none
ata1: enabled=true, ioaddr1=0x170, ioaddr2=0x370, irq=15
ata1-master: type=none
ata1-slave: type=none
ata2: enabled=false
ata3: enabled=false
optromimage1: file=none
```

```
optromimage2: file=none
optromimage3: file=none
optromimage4: file=none
optramimage1: file=none
optramimage2: file=none
optramimage3: file=none
optramimage4: file=none
pci: enabled=1, chipset=i440fx, slot1=none, slot2=none, slot3=none, slot4=none, s
vga: extension=vbe, update_freq=5, realtime=1, ddc=builtin
cpu: count=1, ips=4000000, model=bx_generic, reset_on_triple_fault=1, cpuid_limit
cpuid: level=6, stepping=3, model=3, family=6, vendor_string="GenuineIntel", bran
cpuid: mmx=true, apic=xapic, simd=sse2, sse4a=false, misaligned_sse=false, sep=tr
#cpuid: movbe=false, adx=false, aes=false, sha=false, xsave=false, xsaveopt=false
#cpuid: 1g_pages=false, pcid=false, fsgsbase=false, smep=false, smap=false, mwait
#cpuid: vmx=1
print_timestamps: enabled=0
port_e9_hack: enabled=0
private_colormap: enabled=0
clock: sync=none, time0=local, rtc_sync=0
# no cmosimage
log: -
logprefix: %t%e%d
debug: action=ignore
info: action=report
error: action=report
panic: action=ask
keyboard: type=mf, serial_delay=250, paste_delay=100000, user_shortcut=none
mouse: type=ps2, enabled=false, toggle=ctrl+mbutton
#sound: waveoutdrv=win, waveout=none, waveindrv=win, wavein=none, midioutdrv=win,
speaker: enabled=true, mode=sound, volume=15
parport1: enabled=true, file=none
parport2: enabled=false
com1: enabled=true, mode=null
com2: enabled=false
com3: enabled=false
com4: enabled=false
```

```
megs: 32
# floppya: 1_44=D:/study/code/nasm/bochs/a.img, status=inserted
ata0-master: type=disk, path="/home/hupeng/bin/bochs/c.img", mode=flat
boot: disk
```

准备硬盘镜像

1. 安装 nasm

```
$ sudo apt install nasm
```

2. 编写启动引导代码

```
$ cat boot.asm

org 07c00h
mov ax, cs
mov ds, ax
mov es, ax
call DispStr
jmp $

DispStr:
mov ax, BootMessage
mov bp, ax
mov cx, 16
mov ax, 01301h
mov bx, 000ch
mov dl, 0
int 10h
ret

BootMessage: db "Hello, OS"
times 510 - ($ - $$) db 0
dw 0xaa55
```

3. 编译成二进制文件

```
$ nasm boot.asm -o boot.bin
```

4. 制作软盘镜像 （可选）

```
→ ~/bin/bochs/bin/bximage
========================================================================
                                bximage
  Disk Image Creation / Conversion / Resize and Commit Tool for Bochs
         $Id: bximage.cc 14091 2021-01-30 17:37:42Z sshwarts $
========================================================================

1. Create new floppy or hard disk image
2. Convert hard disk image to other format (mode)
3. Resize hard disk image
4. Commit 'undoable' redolog to base image
5. Disk image info

0. Quit

Please choose one [0] 1

Create image

Do you want to create a floppy disk image or a hard disk image?
Please type hd or fd. [hd] fd

Choose the size of floppy disk image to create.
Please type 160k, 180k, 320k, 360k, 720k, 1.2M, 1.44M, 1.68M, 1.72M, or 2.88M.
 [1.44M]

What should be the name of the image?
[a.img]

Creating floppy image 'a.img' with 2880 sectors

The following line should appear in your bochsrc:
  floppya: image="a.img", status=inserted
```

此时在当前目录下就生成了一个 `a.img` 的软盘镜像。

把启动代码写入软盘镜像

```
$ dd if=boot.bin of=a.img bs=512 count=1 conv=notrunc
```

然后通过修改配置文件，就可以使用软盘启动 `bochs` 了。

5. 制作硬盘镜像

```
→ ~/bin/bochs/bin/bximage
========================================================================
                                bximage
  Disk Image Creation / Conversion / Resize and Commit Tool for Bochs
         $Id: bximage.cc 14091 2021-01-30 17:37:42Z sshwarts $
========================================================================

1. Create new floppy or hard disk image
2. Convert hard disk image to other format (mode)
3. Resize hard disk image
4. Commit 'undoable' redolog to base image
5. Disk image info

0. Quit

Please choose one [0] 1

Create image

Do you want to create a floppy disk image or a hard disk image?
Please type hd or fd. [hd]

What kind of image should I create?
Please type flat, sparse, growing, vpc or vmware4. [flat]

Choose the size of hard disk sectors.
Please type 512, 1024 or 4096. [512]

Enter the hard disk size in megabytes, between 10 and 8257535
[10]

What should be the name of the image?
[c.img]

Creating hard disk image 'c.img' with CHS=20/16/63 (sector size = 512)

The following line should appear in your bochsrc:
  ata0-master: type=disk, path="c.img", mode=flat
```

此时在当前目录下就生成了一个 `c.img` 的硬盘镜像。

把启动代码写入软盘镜像

```
$ dd if=boot.bin of=c.img bs=512 count=1 conv=notrunc
```

然后通过修改配置文件，就可以使用软盘启动 `bochs` 了。

运行调试

1. 运行

```
$ ~/bin/bochs/bin/bochs -f ~/workspace/bochs-2.7/bochrc.bxrc
```

2. 调试

```
$ gdb --args ~/bin/bochs/bin/bochs -f ~/workspace/bochs-2.7/bochrc.bxrc
```

最后，就可以愉快的玩耍了。