

起始	结 束	大 小	用 途
FFFF0	FFFFF	16B	BIOS 入口地址，此地址也属于 BIOS 代码，同样属于顶部的 640KB 字节。只是为了强调其入口地址才单独贴出来。此处 16 字节的内容是跳转指令 <code>jmp f000: e05b</code>
F0000	FFFEF	64KB-16B	系统 BIOS 范围是 F0000~FFFFF 共 640KB，为说明入口地址，将最上面的 16 字节从此处去掉了，所以此处终止地址是 0xFFFEF
C8000	FFFFF	160KB	映射硬件适配器的 ROM 或内存映射式 I/O
C0000	C7FFF	32KB	显示适配器 BIOS
B8000	BFFFF	32KB	用于文本模式显示适配器
B0000	B7FFF	32KB	用于黑白显示适配器
A0000	AFFFF	64KB	用于彩色显示适配器
9FC00	9FFFF	1KB	EBDA (Extended BIOS Data Area) 扩展 BIOS 数据区
7E00	9FBFF	622080B 约 608KB	可用区域
7C00	7DFF	512B	MBR 被 BIOS 加载到此处，共 512 字节
500	7BFF	30464B 约 30KB	可用区域
400	4FF	256B	BIOS Data Area (BIOS 数据区)
000	3FF	1KB	Interrupt Vector Table (中断向量表)

-fd 创建软盘。

-hd 创建硬盘。

-mode 创建硬盘的类型，有 flat、sparse、growing 三种。

-size 指创建多大的硬盘，以 MB 为单位。

-q 以静默模式创建，创建过程中不会和用户交互。

按照上面的帮助，那咱们就开工啦，如图 1-6 所示。

```
bin/bximage -hd -mode="flat" -size=60 -q hd60M.img
```

1. 制作硬盘

```
1 bximage -hd=60M -mode=create -q disk.img
```

2. bochsrc

```
1 # Bochs配置文件
2
3 # 机器内存: 32MB
4 megs: 32
5
```

```
6 # 对应真实机器的bios
7 romimage: file=$BXSHARE/BIOS-bochs-latest, options=fastboot
8 # 对应真实机器的VGA bios
9 vgaromimage: file=$BXSHARE/VGABIOS-lgpl-latest
10
11 # 启动方式
12 boot: disk
13
14 # 关闭鼠标
15 mouse: enabled=0
16
17 #keyboard: type=mf, serial_delay=250
18 keyboard: keymap=$BXSHARE/keymaps/x11-pc-de.map
19
20 # 硬盘设置
21 ata0: enabled=1, ioaddr1=0x1f0, ioaddr2=0x3f0, irq=14
22 ata0-master: type=disk, path="disk.img", cylinders=20, heads=16, spt=63
```

3. 编译代码

```
1 nasm -o mbr.bin mbr.S
```

4. 写入硬盘

```
1 dd if=./mbr.bin of=${v_image} bs=512 count=1 conv=notrunc
```

5. 运行 bochs -f bochsrc, 在控制台输入 c(continue), 即可运行