

华东师范大学数据科学与工程学院实验报告

课程名称：操作系统

年级：2021 级

上机实践成绩：

指导教师：翁楚良

姓名：唐小卉

学号：10215501437

上机实践名称：

上机实践日期：2023.5.1

上机实践编号：

组号：

上机实践时间：2023.5.1

一、实验目的

1. 熟悉类 UNIX 系统的 I/O 设备管理
2. 熟悉 MINIX 块设备驱动
3. 熟悉 MINIX RAM 盘

二、实验任务

1. 在 MINIX3 中安装一块 X MB 大小的 RAM 盘（minix 中已有 6 块用户可用 RAM 盘，7 块系统保留 RAM 盘），可以挂载并且存取文件操作。
2. 测试 RAM 盘和 DISK 盘的文件读写速度，分析其读写速度差异原因（可用图表形式体现在实验报告中）。

三、使用环境

物理机：Windows10

虚拟机：Minix3

虚拟机软件：Vmware

代码编辑：VScode

物理机与虚拟机文件传输：FileZilla

四、实验过程

增加 RAM 盘：

1. 修改/usr/src/minix/drivers/storage/memory/memory.c，增加默认的用户 RAM 盘数：RAMDISKS=7。

```
31
32 #include <machine/vm.h>
33
34 #include "local.h"
35
36 /* ramdisks (/dev/ram*) */
37 #define RAMDISKS 7
```

2. 重新编译内核，重启 reboot。

Make build

Reboot

PS.这里要在/usr/src/中进行编译（MakeFile）

3.创建设备 `mknod /dev/myram b 1 13`，查看设备是否创建成功输入 `ls /dev/ | grep ram`。

```
# mknod /dev/myram b 1 13
# 3.ls /dev/ | grep ram
3.ls: not found
# ls /dev/ | grep ram
myram
ram
ram0
ram1
ram2
ram3
ram4
ram5
#
```

4.实现 `buildmyram` 初始化工具（用于分配容量）。

4.1 参考 `/usr/src/minix/commands/ramdisk/ramdisk.c`，实现 `buildmyram.c`，但是需要将 KB 单位修改成 MB。

```
* ramdisk.c
1
2#include <minix/paths.h>
3
4#include <sys/ioc_memory.h>
5#include <stdio.h>
6#include <fcntl.h>
7#include <stdlib.h>
8
9int main(int argc, char *argv[])
10{
11    int fd;
12    signed long size;
13    char *d;
14
15    if(argc < 2 || argc > 3) {
16        fprintf(stderr, "usage: %s <size in MB> [device]\n",
17            argv[0]);
18        return 1;
19    }
20
21    d = argc == 2 ? _PATH_RAMDISK : argv[2];
22    if((fd=open(d, O_RDONLY)) < 0) {
23        perror(d);
24        return 1;
25    }
26    # 把宏改成1024*1024 (MB)
27    #define KFACTOR 1048576
28    size = atol(argv[1])*KFACTOR;
29
30    if(size < 0) {
31        fprintf(stderr, "size should be non-negative.\n");
32        return 1;
33    }
34
35    if(ioctl(fd, MIOCRAMSIZE, &size) < 0) {
36        perror("MIOCRAMSIZE");
37        return 1;
38    }
39
40    fprintf(stderr, "size on %s set to %ldMB\n", d, size/KFACTOR);
41
42    return 0;
43}
44
45
```

在 Makefile 中添加响应条目

```
* Makefile
1 PROG= ramdisk
2 PROG= buildmyram
3 MAN=
4
5 .include <bsd.prog.mk>
6
```

4.2 编译 buildmyram.c 文件，然后执行命令： buildmyram 500 /dev/myram。创建一个 RAM 盘。

```
# clang buildmyram.c -o buildmyram
buildmyram.c:28:6: warning: implicit declaration of function 'ioctl' is
      in C99 [-Wimplicit-function-declaration]
      if(ioctl(fd, MIOCRAMSIZE, &size) < 0) {
      ^
1 warning generated.
# ./buildmyram
usage: ./buildmyram <size in MB> [device]
# buildmyram 500 /dev/myram
size on /dev/myram set to 500MB
```

PS.这里遇到了问题，最开始发生了 segmentation fault，原因是我的虚拟机的内存为 512MB，而我此时的 size 是 500MB，虚拟机不可能将几乎所有的内存都分给 ram。这里的改进方法是将 size 减小，或者将虚拟机内存调高。

5.在 ram 盘上创建内存文件系统，mkfs.mfs /dev/myram。

6.将 ram 盘挂载到用户目录下，mount /dev/myram /root/myram,查看是否挂在成功：输入 df。

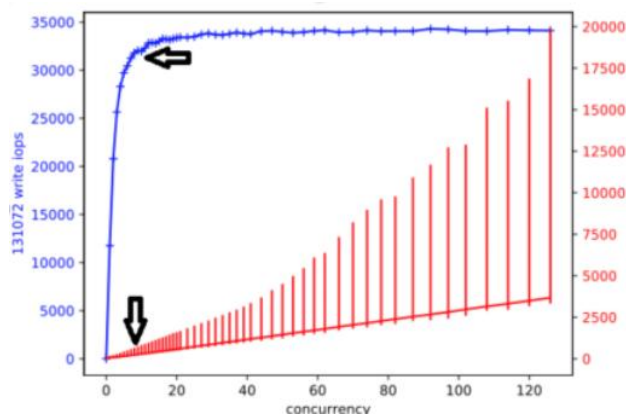
```
# mount /dev/myram /root/myram
/dev/myram is mounted on /root/myram
# df
Filesystem      512-blocks      Used        Avail %Cap Mounted on
/dev/myram      1024000         16088       1007912    1% /root/myram
/dev/c0d0p0s0   262144          84472       177672    32% /
none            0               0           0        100% /proc
/dev/c0d0p0s2   33566464        3664680     29901784   10% /usr
/dev/c0d0p0s1   8114176          84968       8029208    1% /home
none            0               0           0        100% /sys
```

PS.这里需要在/root 中建立 myram 文件夹

性能测试

1. RAM 盘和 Disk 盘的性能测试中，需要采用多进程并发的同步读写，并发数要增加到设备接近“饱和”状态（吞吐量难以继续提升，但是 I/O 延时恶化）。在出现饱和前，总吞吐量随着并发数线性增长。

计算公式：总吞吐量=总文件大小/执行时间



蓝色代表的是吞吐，红色代表的是 I/O 时延。随着并发度的增加，吞吐是先增加，后趋于稳定(甚至恶化降低)。但是 I/O 时延会增加。

Disk 随机写:

```
blocksize_KB=0.0625KB,speed=0.434193MB/s
blocksize_KB=0.2500KB,speed=17.089844MB/s
blocksize_KB=1.0000KB,speed=29.213408MB/s
blocksize_KB=4.0000KB,speed=44.389205MB/s
blocksize_KB=16.0000KB,speed=39.772727MB/s
blocksize_KB=64.0000KB,speed=61.188811MB/s
```

Ram 随机写:

```
blocksize_KB=0.0625KB=64B,speed=2.277431MB/s
blocksize_KB=0.2500KB=256B,speed=18.495502MB/s
blocksize_KB=1.0000KB=1024B,speed=58.829066MB/s
blocksize_KB=4.0000KB=4096B,speed=130.208333MB/s
blocksize_KB=16.0000KB=16384B,speed=213.456284MB/s
blocksize_KB=64.0000KB=65536B,speed=256.990132MB/s
```

Disk 顺序写:

```
blocksize_KB=0.0625KB=64B,speed=11.097301MB/s
blocksize_KB=0.2500KB=256B,speed=35.297440MB/s
blocksize_KB=1.0000KB=1024B,speed=141.189759MB/s
blocksize_KB=4.0000KB=4096B,speed=165.636042MB/s
blocksize_KB=16.0000KB=16384B,speed=244.458931MB/s
blocksize_KB=64.0000KB=65536B,speed=235.626767MB/s
```

Ram 顺序写:

```
blocksize_KB=0.0625KB=64B,speed=12.946851MB/s
blocksize_KB=0.2500KB=256B,speed=34.179688MB/s
blocksize_KB=1.0000KB=1024B,speed=136.718750MB/s
blocksize_KB=4.0000KB=4096B,speed=414.299242MB/s
blocksize_KB=16.0000KB=16384B,speed=597.677596MB/s
blocksize_KB=64.0000KB=65536B,speed=639.619883MB/s
```

Disk 随机读:

```
blocksize_KB=0.0625KB=64B,speed=1.220703MB/s
blocksize_KB=0.2500KB=256B,speed=25.893703MB/s
blocksize_KB=1.0000KB=1024B,speed=51.398026MB/s
blocksize_KB=4.0000KB=4096B,speed=96.280810MB/s
blocksize_KB=16.0000KB=16384B,speed=182.291667MB/s
blocksize_KB=64.0000KB=65536B,speed=238.549618MB/s
```

Ram 随机读:


```
blocksize_KB=0.0625KB,speed=6.376807MB/s  
blocksize_KB=0.2500KB,speed=34.179688MB/s  
blocksize_KB=1.0000KB,speed=103.574811MB/s  
blocksize_KB=4.0000KB,speed=546.875000MB/s  
blocksize_KB=16.0000KB,speed=1093.750000MB/s  
blocksize_KB=64.0000KB,speed=1877.682403MB/s
```

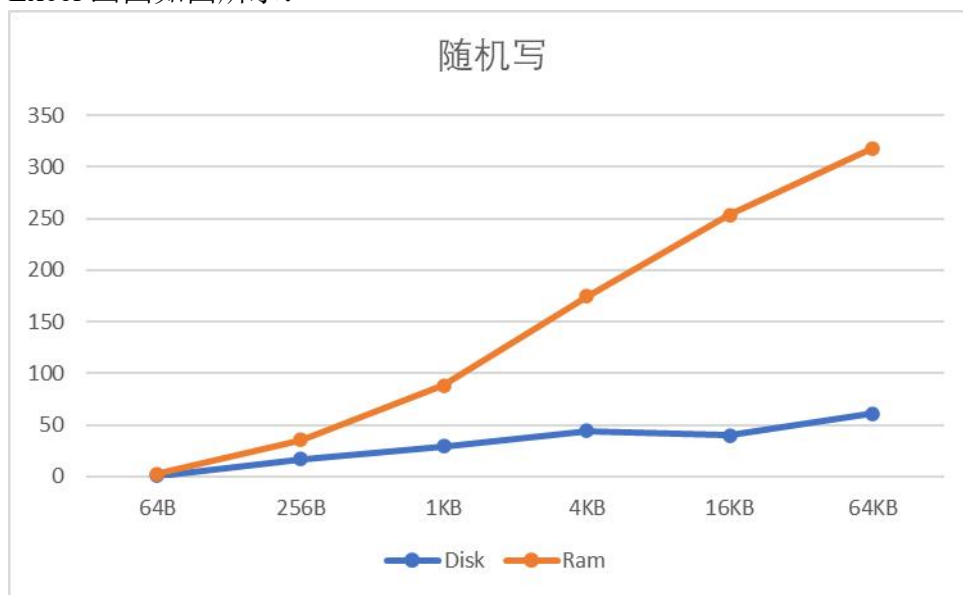
Disk 顺序读:

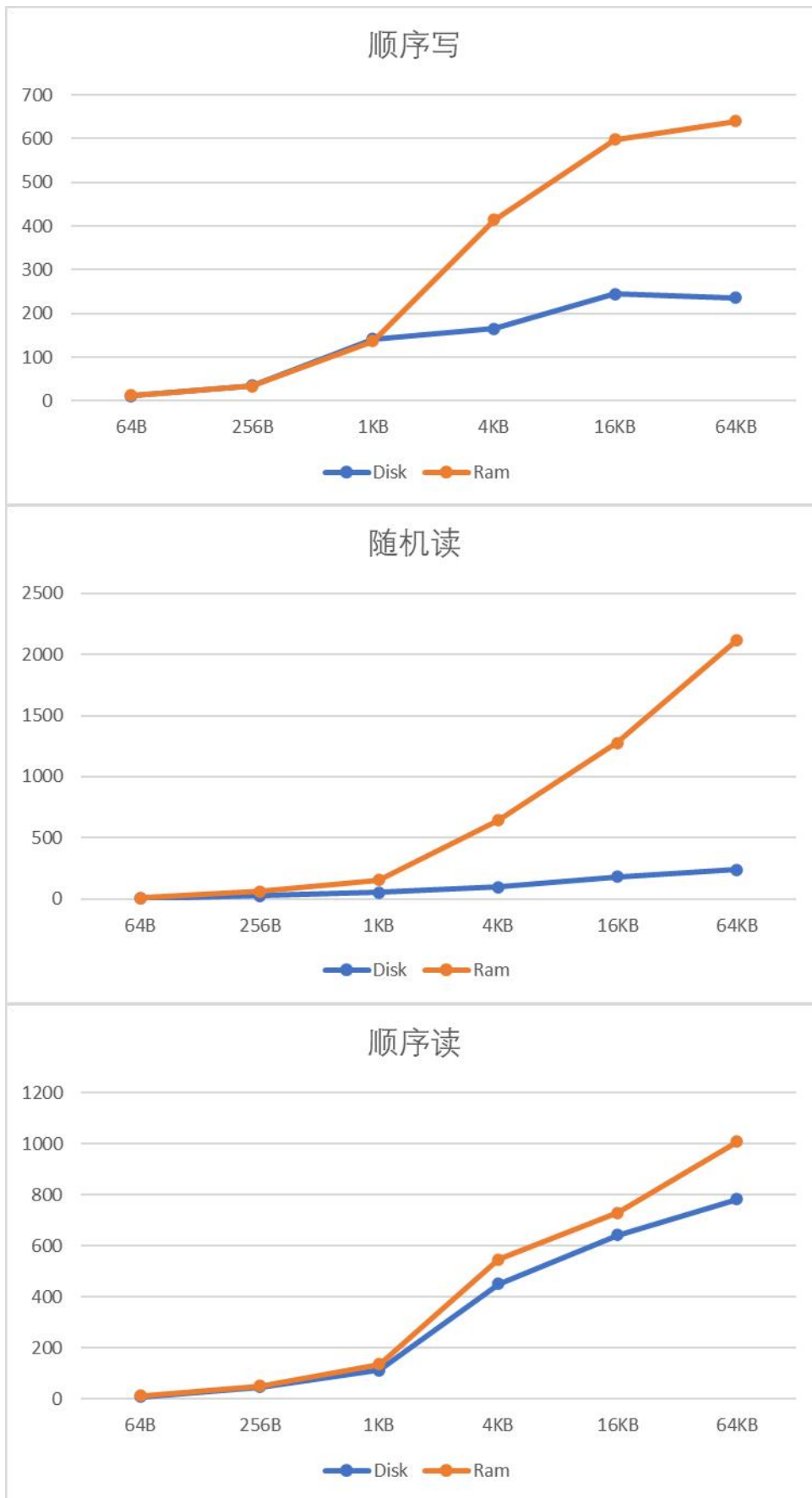
```
blocksize_KB=0.0625KB=64B, speed=8.877841MB/s  
blocksize_KB=0.2500KB=256B, speed=46.875000MB/s  
blocksize_KB=1.0000KB=1024B, speed=112.951807MB/s  
blocksize_KB=4.0000KB=4096B, speed=451.807229MB/s  
blocksize_KB=16.0000KB=16384B, speed=643.776824MB/s  
blocksize_KB=64.0000KB=65536B, speed=782.268579MB/s
```

Ram 顺序读:

```
blocksize_KB=0.0625KB,speed=12.946851MB/s  
blocksize_KB=0.2500KB,speed=51.787405MB/s  
blocksize_KB=1.0000KB,speed=136.718750MB/s  
blocksize_KB=4.0000KB,speed=546.875000MB/s  
blocksize_KB=16.0000KB,speed=729.166667MB/s  
blocksize_KB=64.0000KB,speed=1008.064516MB/s
```

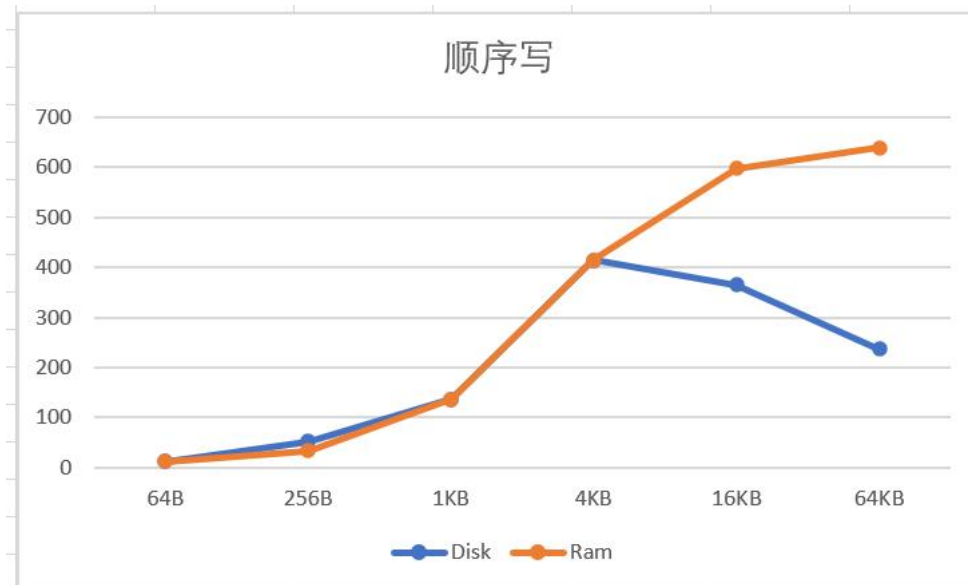
Excel 画图如图所示:





PS.第一次实验的时候，我的 Disk 顺序读和 Disk 顺序写速率没有按照我的预期缓慢增

加，而是出现了转折点（如图所示）



在询问助教后得知，这并不是代码错误的体现，而是因为读写次数、并发度等等变量控制的原因。在变量赋值过大的时候，速率可能会达到瓶颈，从而出现了大幅度下降的现象存在。同时我认为，由于正常情况下速率会平缓上升知道趋于平稳，但偶尔因为误差导致的小幅度下降是正常现象。所以我不断尝试了很多参数，择优画出了现在的 Excel 图像（四舍五入保留两位小数）。每一张图的参数可能会有大大小小的差异，所以横向的比对是不合理的，只能在同一参数下进行比较。

五、总结

在本实验中，我们在 Minix3 系统下分别划分了一块 DISK 盘空间与 RAM 盘空间，并通过一系列不同读写方式的组合测试了 DISK 盘和 RAM 盘的读写速度与特性，更加直观的认识了系统对于 RAM 和 DISK 存储媒介的不同管理方式及其在系统构架中的巨大传输速度差异，从而感受到了现代计算机系统构架的合理性。