

设备管理例题

同济大学计算机系操作系统课程作业

邓蓉

例题 1、

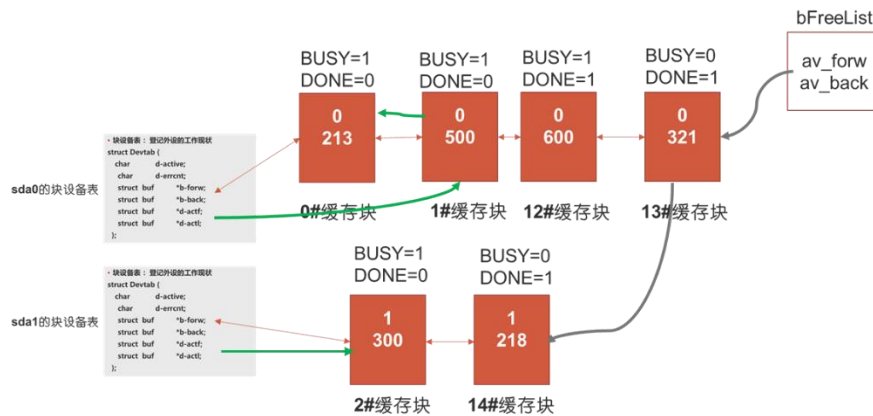


图 3、缓存队列。灰色，自由缓存队列；红色，设备缓存队列；绿色，IO 请求队列

【参考答案】

1、写出系统的自由缓存队列，磁盘 sda0，sda1 的设备缓存队列和 IO 请求队列

● 自由缓存队列：<0,321>→<1,218>

● sda0 的设备缓存队列：<0,213>→<0,500>→<0,600>→<0,321>

● sda0 的 IO 请求队列：<0,500>→<0,213>

● sda1 的设备缓存队列：<1,300>→<1,218>

● sda1 的 IO 请求队列：<1,300>

2、sda0 正在进行的读操作：

● 内存地址：Buffer[1]

● 扇区：500

● IO 完成时，磁盘中断处理程序

■ 会启动新的 IO 请求吗？会，IO 213#扇区

■ 会唤醒进程吗？会，因为读操作是同步的。

3、sda1 正在进行的写操作：

● 内存地址：Buffer[2]

● 扇区：300

● IO 完成时，磁盘中断处理程序

■ 会启动新的 IO 请求吗？一定不会，执行完此 IO 后 IO 请求队列为空

■ 会唤醒进程吗？一般不会，因为通常基于缓存的写操作是异步的。但如果有进程睡眠等待自由缓存，唤醒它。牢记第一种，通常的情况。

例题 2、T0 时刻，PA、PB 进程先后访问文件 A，PA read 4#字节，PB write 4#字节。已知文件 A 的 0#逻辑块（文件的 0#~511#字节）存放在 666#扇区。T0 时刻缓存不命中。自由缓存队列不空，所有自由缓存不脏（不带延迟写标识），队首缓存块 Buffer[7]。

1、请分析如下时刻进程 PA，PB 的调度状态 和 Buffer[i]的使用状态。

- PA 执行 read 系统调用。

缓存不命中。Buffer[7]是 LRU 自由缓存，且不脏，分配用来装新的数据块（666#扇区）。刷新 m_buf[7]：dev=0, blkno=666, B_READ=1, B_DONE=0，送 IO 请求队列之后，PA 执行 IOWait 函数入睡：

 - sleep(& m_buf[7], -50)，高优先级睡眠。等待 IO 完成，也就是 m_buf[7]的 B_DONE 变 1。

Buffer[7]状态：上锁(B_BUSY==1)，在 IO 请求队列；数据不可用 (B_DONE==0)。进程 PA 持锁。
- PB 执行 write 系统调用

缓存命中，PB 执行 GetBlk 函数时入睡：

 - sleep(& m_buf[7], -50)，高优先级睡眠。等待持锁进程 PA 执行 Brelse()解锁，也就是 Buf[7] 的 B_BUSY 变 1。B_WANTED 置 1。

Buffer[7] 状态：上锁(B_BUSY==1)，在 IO 请求队列；数据不可用 (B_DONE==0)。进程 PA 持锁。有进程等待使用其中的数据 (B_WANTED==1)。
- 666#扇区 IO 完成

中断处理程序执行 IODone()函数，m_buf[7] B_DONE=1 置 1，唤醒 PA 和 PB。PA、PB 就绪，之后依次上台运行。

Buffer[7] 状态：上锁(B_BUSY==1)，不在 IO 请求队列，在设备缓存队列，不在自由缓存队列；数据可用 (B_DONE==1)。进程 PA 持锁。有进程等待使用其中的数据 (B_WANTED==1)。

 - 若 PA 先上台运行。系统先后发生以下行为。
 - ◆ B_DONE==1，PA 执行 IOmove 将 4#字节复制进用户空间。解锁缓存 (B_BUSY=0, B_WANTED=0)，送自由缓存队列队尾；
 - ◆ PB 上台运行。B_BUSY==0，PB 锁住缓存，先读后写，将新数据写入 Buffer[7] 4#字节后，置延迟写标记（脏标记）B_DELWR。解锁缓存，标记缓存块中数据可用（是最新版本）B_DONE=1，送自由缓存队列队尾。（没 IO 的）
 - 若 PB 先上台运行。系统先后发生以下行为。
 - ◆ B_BUSY==1。PB，sleep(& m_buf[7], -50)再次入睡。置 1 Buf[7]的 B_WANTED 标识。
 - Buffer[7] 状态：上面绿的。
 - ◆ PA 上台运行。B_DONE==1，PA 执行 IOmove 将 4#字节复制进用户空间。解锁缓存 (B_BUSY=0, B_WANTED=0)，送自由缓存队列队尾；B_WANTED==1，唤醒 PB。
 - ◆ 上面紫的。
- PA 进程能够读到 PB 进程写入的新数据吗？

不能。因为，IO 完成时，一定是 PA 先使用缓存块中的数据。

例题 3: 进程读磁盘文件, 存在不入睡的可能吗? 何时? 存在。需要读入的数据命中自由缓存时, 进程无需入睡。

例题 4: 在 UNIX V6++ 中, 试说明缓存控制块 Buf 有无可能, 在什么样的条件下出现下列情况:

- (1) 同时处在自由 Buf 和一个设备 Buf 队列中;
可能, Brelse 释放的自由缓存块
- (2) 同时处在某一设备 Buf 队列和 I/O 请求队列中;
可能, IO 操作尚未完成的所有缓存块都是这样的
- (3) 只处在某一设备 Buf 队列中;
可能。IO 完成 (B_DONE 是 1) 但 IOmove 还没有完成的缓存块。
- (4) 只处在 I/O 请求队列中, 不在设备缓存队列里。
可能。负责传送进程图像的 SwBuf。
- (5) 同时出现在自由 Buf、某一设备的 Buf 队列和 I/O 请求队列中;
不可能, 自由缓存不可能在 IO 请求队列, 反之亦然。
- (6) 同时出现在一类设备的 Buf 队列、另一类设备的 I/O 请求队列中;
不可能, 一个缓存块不可能分配给 2 个设备。
- (7) 同时出现在自由 Buf 队列和一个设备的 I/O 请求队列中。
不可能。同第 5 题。

例题 4、

1、如果全是读操作, 看这个例题

- 时刻 T1, 下述数据块序列缓存不命中。请比较使用缓存池的 IO 和 未使用缓存池的 IO。性能指标
(1) 完成 IO 请求的平均耗时 (2) 完成整个 IO 请求序列系统的总耗时 (3) IO 总次数

已知: 将 IO 请求放入 IO 请求队列后至 IO 操作完成, 进程平均需要等待 T。

从缓存复制数据到用户空间 (含上锁、解锁操作), 平均耗时 t。 $T > t$ 。

IO 请求序列 556, 556, 556, 600, 782, 891, 900, 556, 556, 556

未使用缓存池, 数据从磁盘直接复制到用户空间, IO 请求平均耗时 T。

完成整个 IO 请求序列系统的总耗时, $10 * T$ 。 IO, 10 次。

使用缓存池, 数据从磁盘复制到缓存。之后进程从缓存中取用数据, 复制到自己的用户空间。完成整个序列, IO 5 个磁盘数据块。总耗时, $5 * T + 10 * t$ 。 IO, 5 次。

IO 请求平均耗时 $T/2 + t$ 。

完成序列中全部的 IO 请求, 系统需要执行 5 次 (有缓冲池) 10 次 (无缓冲池) 磁盘 IO 操作?
磁盘高速缓存的作用? IO 数据重用。有利于减少 IO 请求次数、提高磁盘吞吐率; 有利于减小 IO 操作的平均耗时