**PROBLEMS**

**CHAPTER 5. INPUT/OUTPUT**

1. 1. Advances in chip technology have made it possible to put an entire controller, includ-ing all the bus access logic, on an inexpensive chip. How does that affect the model of Fig. 1-6?
2. 2. Given the speeds listed in Fig. 5-1, is it possible to scan documents from a scanner and transmit them over an 802.11g network at full speed? Defend your answer.
3. 3. Figure 5-3(b) shows one way of having memory-mapped I/O even in the presence of separate buses for memory and I/O devices, namely, to first try the memory bus and if that fails try the I/O bus. A clever computer science student has thought of an im-provement on this idea: try both in parallel, to speed up the process of accessing I/O devices. What do you think of this idea?
4. 4. Explain the tradeoffs between precise and imprecise interrupts on a superscalar machine.
5. 5. A DMA controller has five channels. The controller is capable of requesting a 32-bit word every 40 nsec. A response takes equally long. How fast does the bus have to be to avoid being a bottleneck?
6. 6. Suppose that a system uses DMA for data transfer from disk controller to main memo-ry. Further assume that it takes *t*1 nsec on average to acquire the bus and *t*2 nsec to transfer one word over the bus (*t*1 >> *t*2). After the CPU has programmed the DMA controller, how long will it take to transfer 1000 words from the disk controller to main memory, if (a) word-at-a-time mode is used, (b) burst mode is used? Assume that com-manding the disk controller requires acquiring the bus to send one word and acknowl-edging a transfer also requires acquiring the bus to send one word.
7. 7. One mode that some DMA controllers use is to have the device controller send the word to the DMA controller, which then issues a second bus request to write to mem-ory. How can this mode be used to perform memory to memory copy? Discuss any advantage or disadvantage of using this method instead of using the CPU to perform memory to memory copy.
8. 8. Suppose that a computer can read or write a memory word in 5 nsec. Also suppose that when an interrupt occurs, all 32 CPU registers, plus the program counter and PSW are pushed onto the stack. What is the maximum number of interrupts per second this ma-chine can process?
9. 9. CPU architects know that operating system writers hate imprecise interrupts. One way to please the OS folks is for the CPU to stop issuing new instructions when an interrupt is signaled, but allow all the instructions currently being executed to finish, then force the interrupt. Does this approach have any disadvantages? Explain your answer.
10. 10. In Fig. 5-9(b), the interrupt is not acknowledged until after the next character has been output to the printer. Could it have equally well been acknowledged right at the start of the interrupt service procedure? If so, give one reason for doing it at the end, as in the text. If not, why not?
11. 11. A computer has a three-stage pipeline as shown in Fig. 1-7(a). On each clock cycle, one new instruction is fetched from memory at the address pointed to by the PC and put into the pipeline and the PC advanced. Each instruction occupies exactly one mem-ory word. The instructions already in the pipeline are each advanced one stage. When an interrupt occurs, the current PC is pushed onto the stack, and the PC is set to the ad-dress of the interrupt handler. Then the pipeline is shifted right one stage and the first instruction of the interrupt handler is fetched into the pipeline. Does this machine have precise interrupts? Defend your answer.
12. 12. A typical printed page of text contains 50 lines of 80 characters each. Imagine that a certain printer can print 6 pages per minute and that the time to write a character to the printer’s output register is so short it can be ignored. Does it make sense to run this printer using interrupt-driven I/O if each character printed requires an interrupt that takes 50 *μ* sec all-in to service?
13. 13. Explain how an OS can facilitate installation of a new device without any need for recompiling the OS.
14. 14. In which of the four I/O software layers is each of the following done.
    1. (a) Computing the track, sector, and head for a disk read.
    2. (b) Writing commands to the device registers.
    3. (c) Checking to see if the user is permitted to use the device.
    4. (d) Converting binary integers to ASCII for printing.
15. 15. A local area network is used as follows. The user issues a system call to write data packets to the network. The operating system then copies the data to a kernel buffer.

Then it copies the data to the network controller board. When all the bytes are safely inside the controller, they are sent over the network at a rate of 10 megabits/sec. The receiving network controller stores each bit a microsecond after it is sent. When the last bit arrives, the destination CPU is interrupted, and the kernel copies the newly arri-ved packet to a kernel buffer to inspect it. Once it has figured out which user the packet is for, the kernel copies the data to the user space. If we assume that each interrupt and its associated processing takes 1 msec, that packets are 1024 bytes (ignore the head-ers), and that copying a byte takes 1 *μ* sec, what is the maximum rate at which one process can pump data to another? Assume that the sender is blocked until the work is finished at the receiving side and an acknowledgement comes back. For simplicity, as-sume that the time to get the acknowledgement back is so small it can be ignored.

1. 16. Why are output files for the printer normally spooled on disk before being printed?
2. 17. How much cylinder skew is needed for a 7200-RPM disk with a track-to-track seek time of 1 msec? The disk has 200 sectors of 512 bytes each on each track.
3. 18. A disk rotates at 7200 RPM. It has 500 sectors of 512 bytes around the outer cylinder. How long does it take to read a sector?
4. 19. Calculate the maximum data rate in bytes/sec for the disk described in the previous problem.
5. 20. RAID level 3 is able to correct single-bit errors using only one parity drive. What is the point of RAID level 2? After all, it also can only correct one error and takes more drives to do so.
6. 21. A RAID can fail if two or more of its drives crash within a short time interval. Suppose that the probability of one drive crashing in a given hour is *p*. What is the probability of a *k*-drive RAID failing in a given hour?
7. 22. Compare RAID level 0 through 5 with respect to read performance, write performance, space overhead, and reliability.
8. 23. How many pebibytes are there in a zebibyte?
9. 24. Why are optical storage devices inherently capable of higher data density than mag-netic storage devices? *Note*: This problem requires some knowledge of high-school physics and how magnetic fields are generated.
10. 25. What are the advantages and disadvantages of optical disks versus magnetic disks?
11. 26. If a disk controller writes the bytes it receives from the disk to memory as fast as it re-ceives them, with no internal buffering, is interleaving conceivably useful? Discuss your answer.
12. 27. If a disk has double interleaving, does it also need cylinder skew in order to avoid missing data when making a track-to-track seek? Discuss your answer.
13. 28. Consider a magnetic disk consisting of 16 heads and 400 cylinders. This disk has four 100-cylinder zones with the cylinders in different zones containing 160, 200, 240. and 280 sectors, respectively. Assume that each sector contains 512 bytes, average seek time between adjacent cylinders is 1 msec, and the disk rotates at 7200 RPM. Calcu-late the (a) disk capacity, (b) optimal track skew, and (c) maximum data transfer rate.
14. 29. A disk manufacturer has two 5.25-inch disks that each have 10,000 cylinders. The newer one has double the linear recording density of the older one. Which disk proper-ties are better on the newer drive and which are the same? Are any worse on the newer one?
15. 30. A computer manufacturer decides to redesign the partition table of a Pentium hard disk to provide more than four partitions. What are some consequences of this change?
16. 31. Disk requests come in to the disk driver for cylinders 10, 22, 20, 2, 40, 6, and 38, in that order. A seek takes 6 msec per cylinder. How much seek time is needed for
    1. (a) First-come, first served.
    2. (b) Closest cylinder next.
    3. (c) Elevator algorithm (initially moving upward).

In all cases, the arm is initially at cylinder 20.

1. 32. A slight modification of the elevator algorithm for scheduling disk requests is to al-ways scan in the same direction. In what respect is this modified algorithm better than the elevator algorithm?
2. 33. A personal computer salesman visiting a university in South-West Amsterdam remark-ed during his sales pitch that his company had devoted substantial effort to making their version of UNIX very fast. As an example, he noted that their disk driver used the elevator algorithm and also queued multiple requests within a cylinder in sector order. A student, Harry Hacker, was impressed and bought one. He took it home and wrote a program to randomly read 10,000 blocks spread across the disk. To his amaze-ment, the performance that he measured was identical to what would be expected from first-come, first-served. Was the salesman lying?
3. 34. In the discussion of stable storage using nonvolatile RAM, the following point was glossed over. What happens if the stable write completes but a crash occurs before the operating system can write an invalid block number in the nonvolatile RAM? Does this race condition ruin the abstraction of stable storage? Explain your answer.
4. 35. In the discussion on stable storage, it was shown that the disk can be recovered to a consistent state (a write either completes or does not take place at all) if a CPU crash occurs during a write. Does this property hold if the CPU crashes again during a recov-ery procedure. Explain your answer.
5. 36. In the discussion on stable storage, a key assumption is that a CPU crash that corrupts a sector leads to an incorrect ECC. What problems might arise in the five crash-recov-ery scenarios shown in Figure 5-27 if this assumption does not hold?
6. 37. The clock interrupt handler on a certain computer requires 2 msec (including process switching overhead) per clock tick. The clock runs at 60 Hz. What fraction of the CPU is devoted to the clock?
7. 38. A computer uses a programmable clock in square-wave mode. If a 500 MHz crystal is used, what should be the value of the holding register to achieve a clock resolution of
   1. (a) a millisecond (a clock tick once every millisecond)?
   2. (b) 100 microseconds?
8. 39. A system simulates multiple clocks by chaining all pending clock requests together as shown in Fig. 5-30. Suppose the current time is 5000 and there are pending clock re-quests for time 5008, 5012, 5015, 5029, and 5037. Show the values of Clock header, Current time, and Next signal at times 5000, 5005, and 5013. Suppose a new (pending) signal arrives at time 5017 for 5033. Show the values of Clock header, Current time and Next signal at time 5023.
9. 40. Many versions of UNIX use an unsigned 32-bit integer to keep track of the time as the number of seconds since the origin of time. When will these systems wrap around (year and month)? Do you expect this to actually happen?
10. 41. A bitmap terminal contains 1600 by 1200 pixels. To scroll a window, the CPU (or controller) must move all the lines of text upward by copying their bits from one part of the video RAM to another. If a particular window is 80 lines high by 80 characters wide (6400 characters, total), and a character’s box is 8 pixels wide by 16 pixels high, how long does it take to scroll the whole window at a copying rate of 50 nsec per byte? If all lines are 80 characters long, what is the equivalent baud rate of the terminal? Putting a character on the screen takes 5 *μ* sec. How many lines per second can be dis-played?
11. 42. After receiving a DEL (SIGINT) character, the display driver discards all output cur-rently queued for that display. Why?
12. 43. A user at a terminal issues a command to an editor to delete the word on line 5 occupy-ing character positions 7 through and including 12. Assuming the cursor is not on line 5 when the command is given, what ANSI escape sequence should the editor emit to delete the word?
13. 44. The designers of a computer system expected that the mouse could be moved at a max-imum rate of 20 cm/sec. If a mickey is 0.1 mm and each mouse message is 3 bytes, what is the maximum data rate of the mouse assuming that each mickey is reported separately?
14. 45. The primary additive colors are red, green, and blue, which means that any color can be constructed from a linear superposition of these colors. Is it possible that someone could have a color photograph that cannot be represented using full 24-bit color?
15. 46. One way to place a character on a bitmapped screen is to use *BitBlt* from a font table. Assume that a particular font uses characters that are 16 × 24 pixels in true RGB color.
    1. (a) How much font table space does each character take?
    2. (b) If copying a byte takes 100 nsec, including overhead, what is the output rate to the screen in characters/sec?
16. 47. Assuming that it takes 2 nsec to copy a byte, how much time does it take to completely rewrite the screen of an 80 character × 25 line text mode memory-mapped screen? What about a 1024 × 768 pixel graphics screen with 24-bit color?
17. 48. In Fig. 5-36 there is a class to *RegisterClass*. In the corresponding X Window code, in Fig. 5-34, there is no such call or anything like it. Why not?
18. 49. In the text we gave an example of how to draw a rectangle on the screen using the Win-dows GDI:

Rectangle(hdc, xleft, ytop, xright, ybottom);

Is there any real need for the first parameter (*hdc*), and if so, what? After all, the coor-dinates of the rectangle are explicitly specified as parameters.

1. 50. A thin-client terminal is used to display a Web page containing an animated cartoon of size 400 pixels × 160 pixels running at 10 frames/sec. What fraction of a 100-Mbps Fast Ethernet is consumed by displaying the cartoon?
2. 51. It has been observed that a thin-client system works well with a 1-Mbps network in a test. Are any problems likely in a multiuser situation? (*Hint*: Consider a large number of users watching a scheduled TV show and the same number of users browsing the World Wide Web.)
3. 52. Describe two advantages and two disadvantages of thin client computing?
4. 53. If a CPU’s maximum voltage, *V* , is cut to *V* /*n*, its power consumption drops to 1/*n*2 of its original value and its clock speed drops to 1/*n* of its original value. Suppose that a user is typing at 1 char/sec, but the CPU time required to process each character is 100 msec. What is the optimal value of *n* and what is the corresponding energy saving in percent compared to not cutting the voltage? Assume that an idle CPU consumes no energy at all.
5. 54. A notebook computer is set up to take maximum advantage of power saving features including shutting down the display and the hard disk after periods of inactivity. A user sometimes runs UNIX programs in text mode, and at other times uses the X Window System. She is surprised to find that battery life is significantly better when she uses text-only programs. Why?
6. 55. Write a program that simulates stable storage. Use two large fixed-length files on your disk to simulate the two disks.
7. 56. Write a program to implement the three disk-arm scheduling algorithms. Write a driver program that generates a sequence of cylinder numbers (0–999) at random, runs the three algorithms for this sequence and prints out the total distance (number of cylin-ders) the arm needs to traverse in the three algorithms.
8. 57. Write a program to implement multiple timers using a single clock. Input for this pro-gram consists of a sequence of four types of commands (S <int>, T, E <int>, P): S <int> sets the current time to <int>; T is a clock tick; and E <int> schedules a signal to occur at time <int>; P prints out the values of Current time, Next signal, and Clock header. Your program should also print out a statement whenever it is time to raise a signal.