5.{2,5,8,11,12,13,15,17,18,19,21}

**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.

37.5mb/s network vs 1 mb/s scanners rate, yes it is possible

**5.** A DMA controller has fiv e 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?

32bit/(40+40)nsec = 40 x 10^7 bit/sec = .8 x 10^7 b/s

**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 machine

can process?

34 words written. Then new PC is +1 I/O = 35 x 5 nsec, 175\*10^-6

1/175\*10^-6 = 5.71428 x 10^7 interrupts/sec

**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 memory

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 address

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.

Yes, the first instruction not fetched is pointed to by stacked PC. No instructions after have been executed and all instructions before have been, therefore precise interrupts. Easy to do on machine with single pipeline, unless instructions are out of order then problems occur.

**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 sec all-in to service?

50 x 80 x 6 = 24,000 char/min = 400 char/sec, 50micsec CPU interrupt, so 20msec. 980 msec available for other use, nearly 2% of CPU used for interrupt.

**13.** Explain how an OS can facilitate installation of a new device without any need for

recompiling the OS.

In unix, a table is indexed by device number, with C struct table entries containing pointers to functions for open, close, read, and write, etc, from device. New entries must be made in this table and pointers used to new device driver.

**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 arrived

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 headers),

and that copying a byte takes 1 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, assume

that the time to get the acknowledgement back is so small it can be ignored.

Packet is copied 4 times, taking 4.1 msec. Two interrupts take 2msec. Transmission is 0.83 msec, 6.93 msec per 1024 bytes. Max data rate being 147,763 bytes/sec, roughly 12% of nominal 10 megabit/sec network capacity.

**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.

Disk rotation 120 rpm, 1000/120msec = 1 rotation. 200 sectors per rotation, 1/200 or 5/120 = 1/24 msec. During 1msec seek, 24 sectors pass so skew should be 24.

**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?

1000/120msec one rotation, 1000/120 \* 1/500 = 1/60 msec

**19.** Calculate the maximum data rate in bytes/sec for the disk described in the previous

problem.

Sector = 1/60msec

60000 sectors/sec \* 512 bytes = 30,720,000 bytes/sec = 29.30 MB/sec

**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?

1-(1-p)^k-kp(1-p)^(k-1)