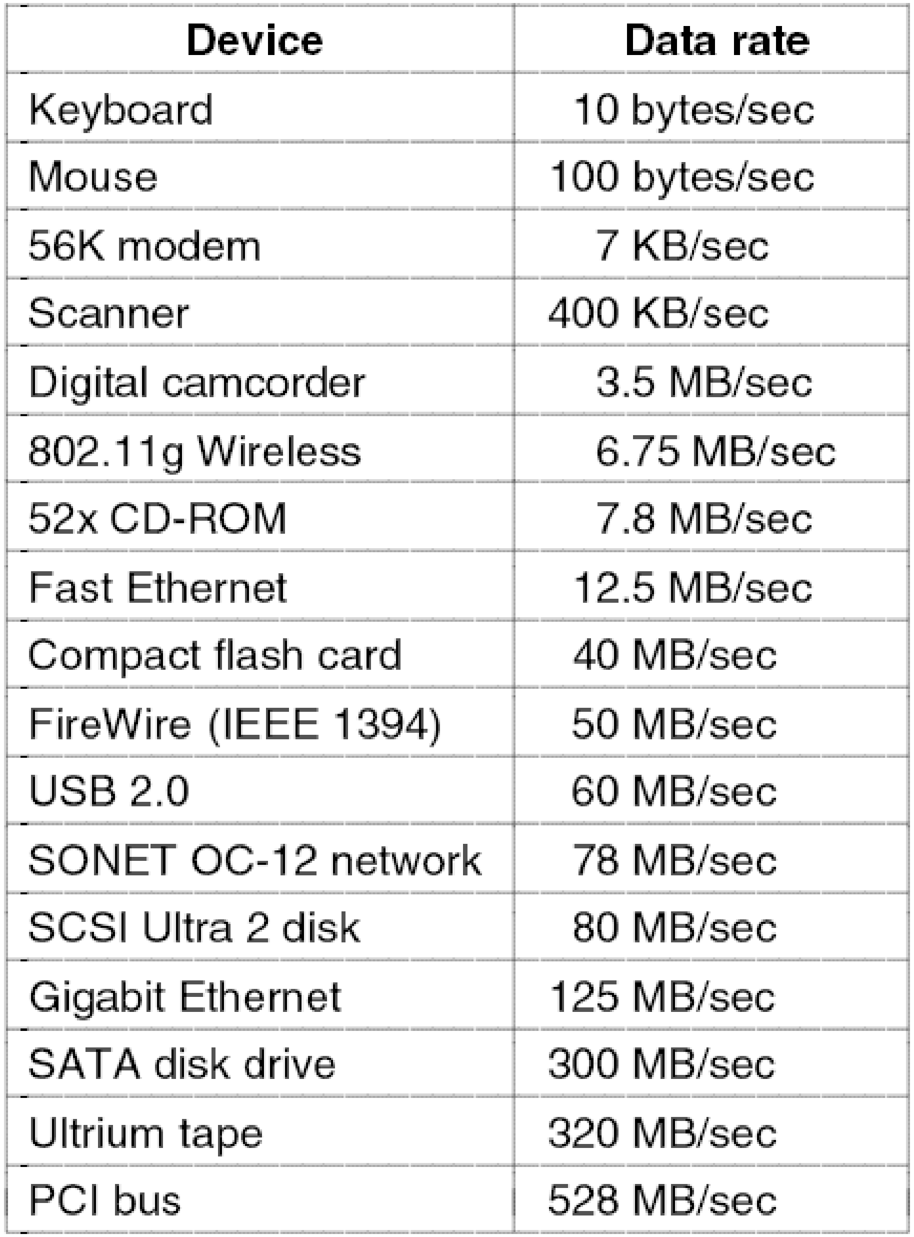
MODERN OPERATING SYSTEMS

Third Edition
ANDREW S. TANENBAUM

Chapter 5
Input/Output

I/O Devices



Device	Data rate
Keyboard	10 bytes/sec
Mouse	100 bytes/sec
56K modem	7 KB/sec
Scanner	400 KB/sec
Digital camcorder	3.5 MB/sec
802.11g Wireless	6.75 MB/sec
52x CD-ROM	7.8 MB/sec
Fast Ethernet	12.5 MB/sec
Compact flash card	40 MB/sec
FireWire (IEEE 1394)	50 MB/sec
USB 2.0	60 MB/sec
SONET OC-12 network	78 MB/sec
SCSI Ultra 2 disk	80 MB/sec
Gigabit Ethernet	125 MB/sec
SATA disk drive	300 MB/sec
Ultrium tape	320 MB/sec
PCI bus	528 MB/sec

Figure 5-1. Some typical device, network, and bus data rates.

Memory-Mapped I/O

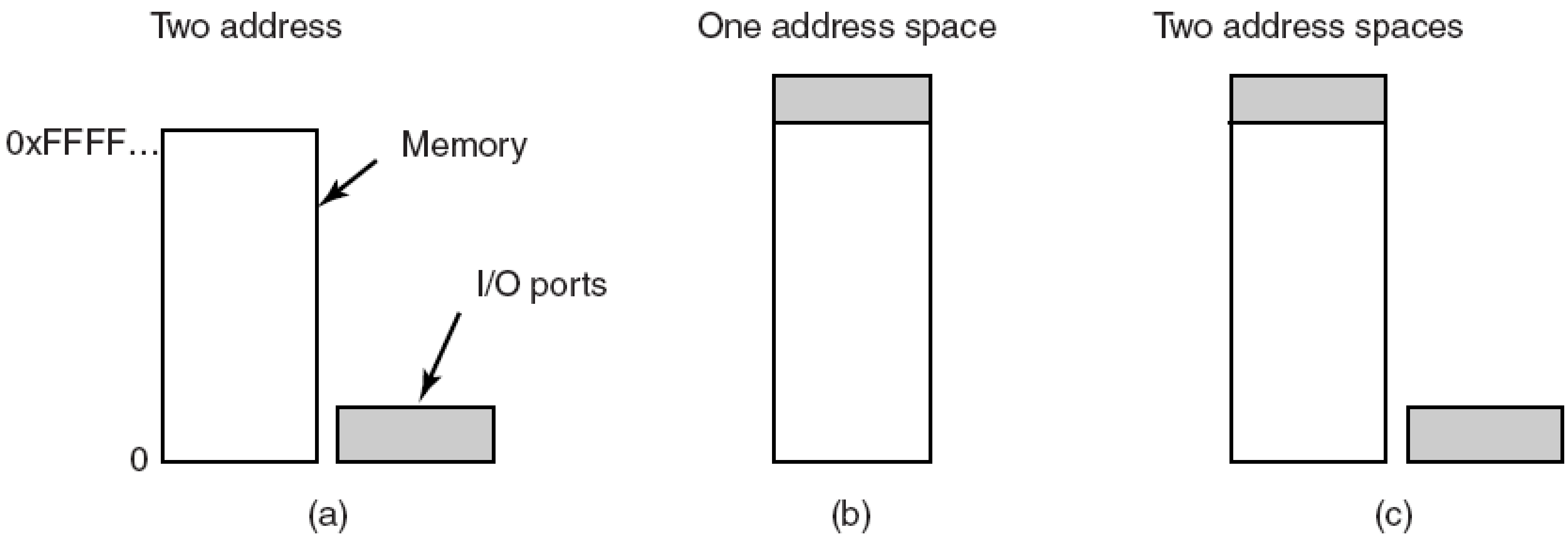


Figure 5-2. (a) Separate I/O and memory space. (b) Memory-mapped I/O. (c) Hybrid.

Direct Memory Access (DMA)

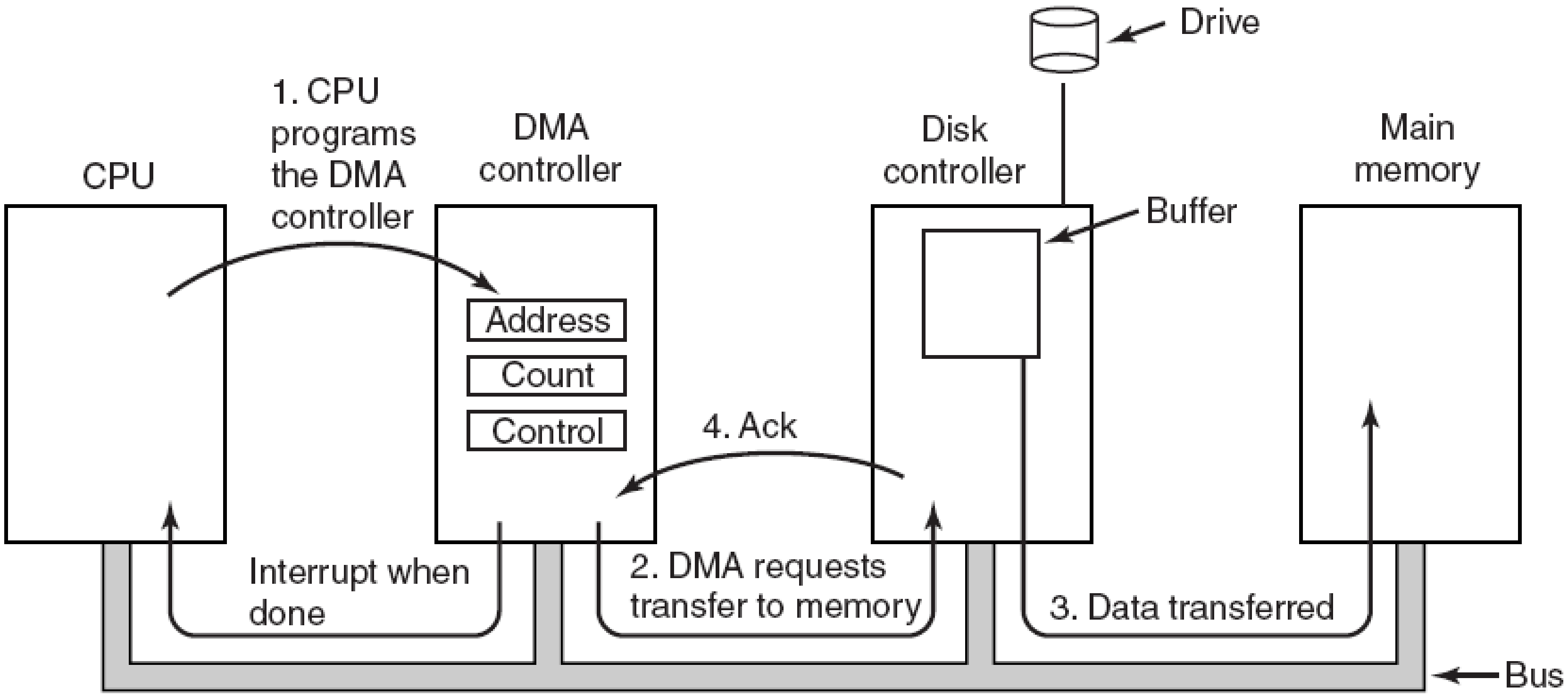


Figure 5-4. Operation of a DMA transfer.

Interrupts Revisited

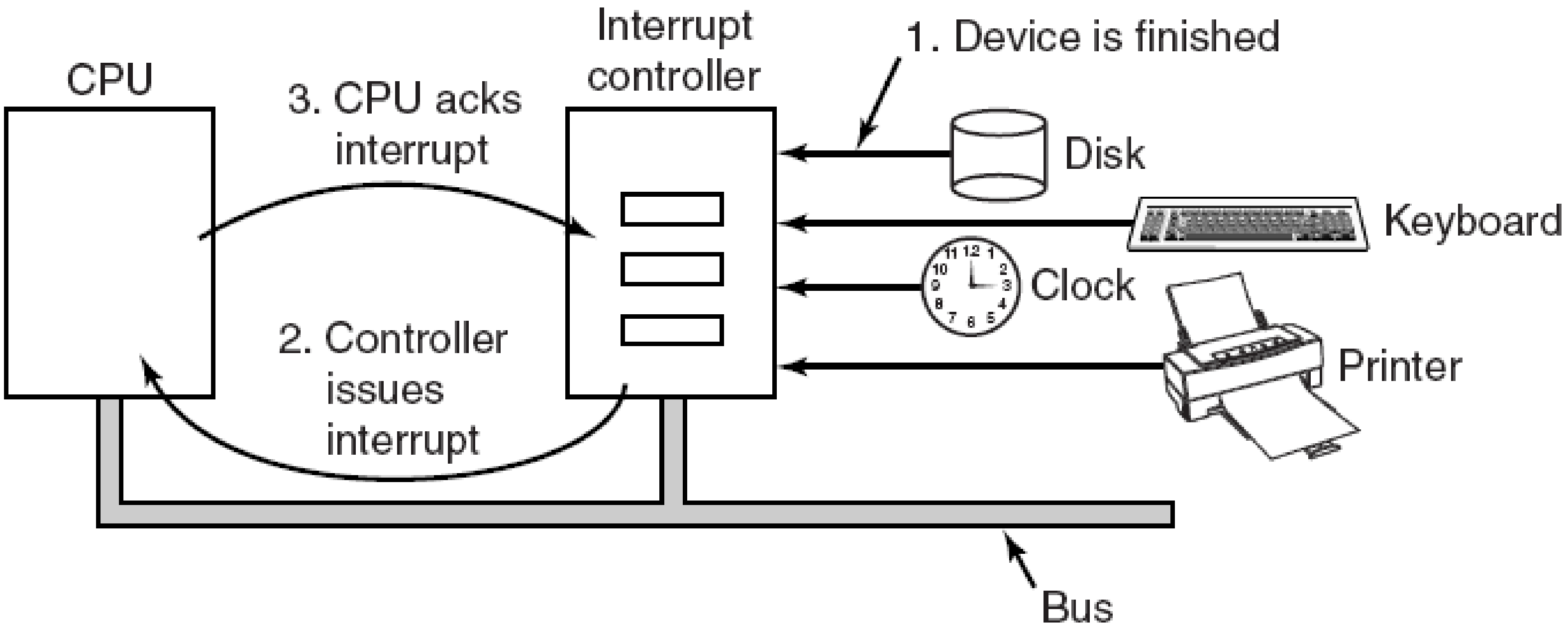


Figure 5-5. How an interrupt happens. The connections between the devices and the interrupt controller actually use interrupt lines on the bus rather than dedicated wires.

Programmed I/O (1)

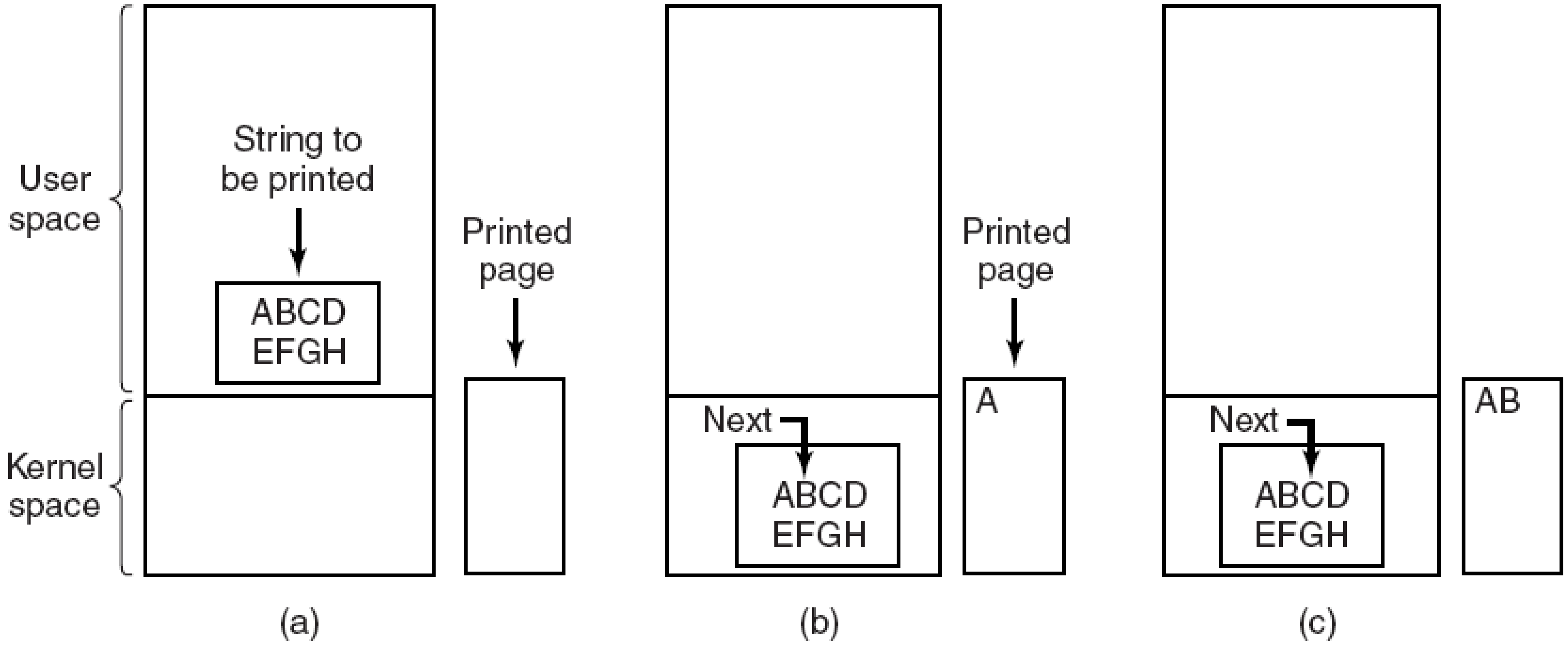
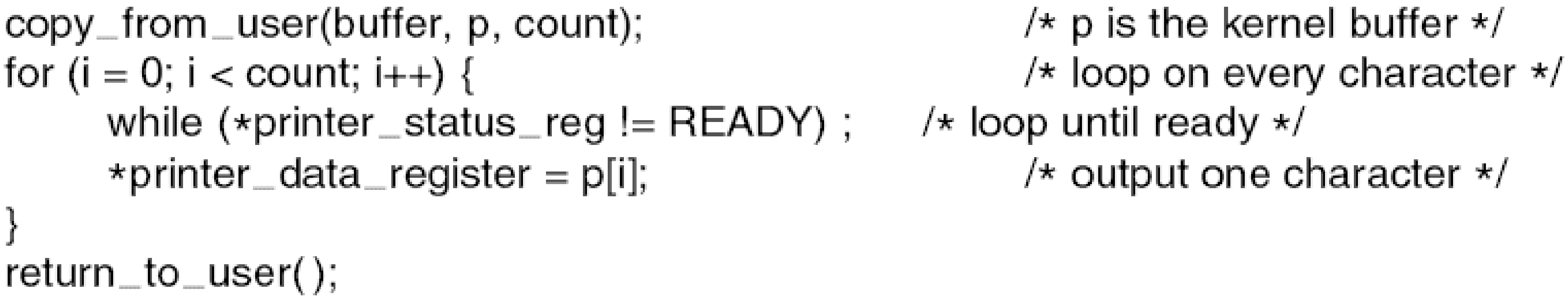


Figure 5-7. Steps in printing a string.

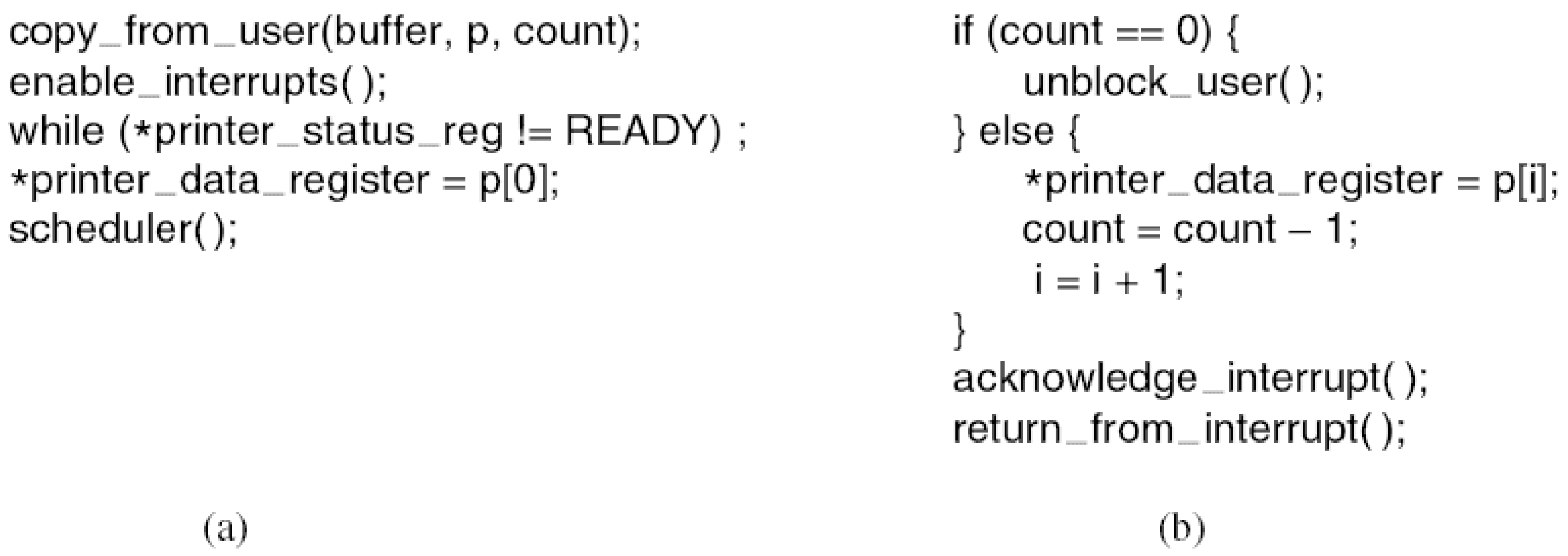
Programmed I/O (2)



```
copy_from_user(buffer, p, count);               /* p is the kernel buffer */
for (i = 0; i < count; i++) {                   /* loop on every character */
     while (*printer_status_reg != READY) ;  /* loop until ready */
     *printer_data_register = p[i];             /* output one character */
}
return_to_user( );
```

Figure 5-8. Writing a string to the printer using programmed I/O.

Interrupt-Driven I/O



```
copy_from_user(buffer, p, count);
enable_interrupts( );
while (*printer_status_reg != READY) ;
*printer_data_register = p[0];
scheduler( );
```

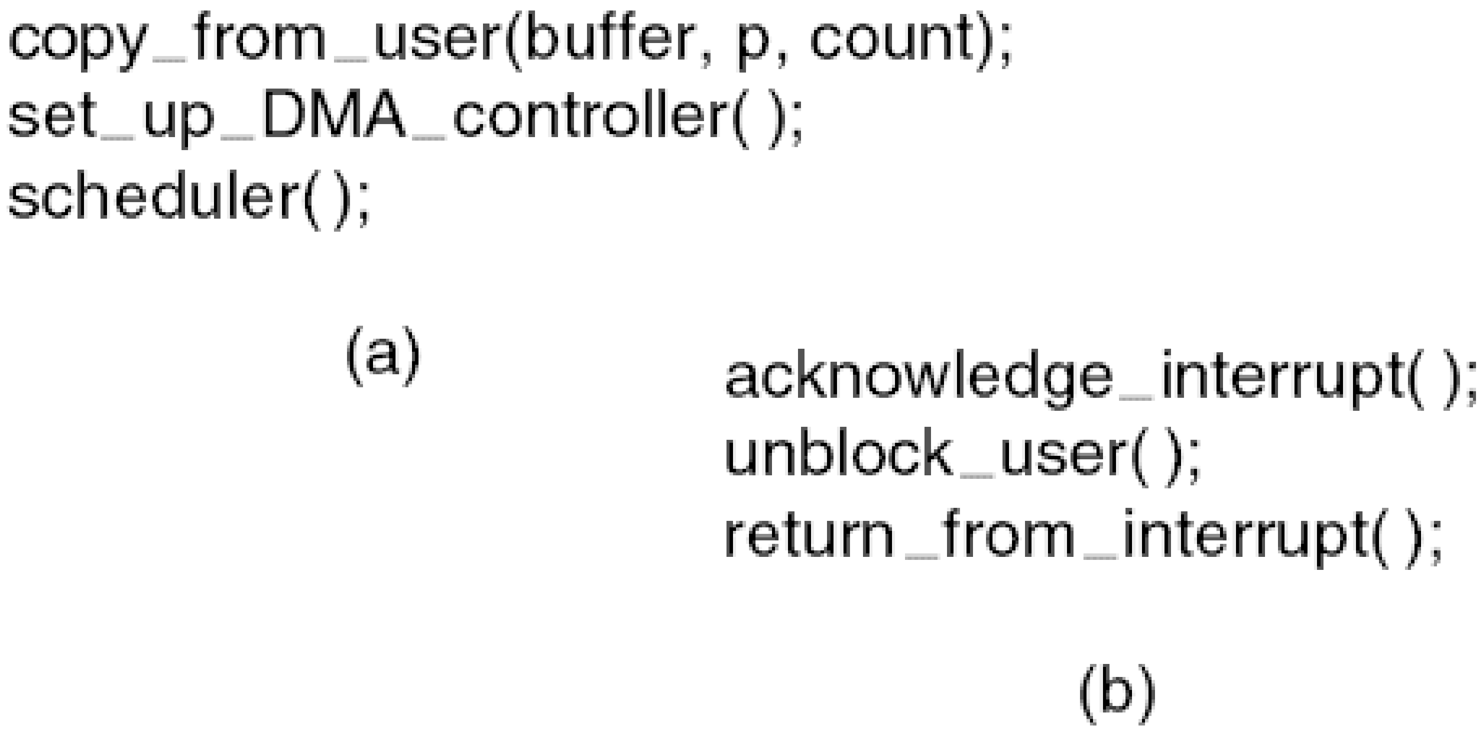
(a)

```
if (count == 0) {
      unblock_user( );
} else {
      *printer_data_register = p[i];
      count = count − 1;
       i = i + 1;
}
acknowledge_interrupt( );
return_from_interrupt( );
```

(b)

Figure 5-9. Writing a string to the printer using interrupt-driven I/O. (a) Code executed at the time the print system call is made. (b) Interrupt service procedure for the printer.

I/O Using DMA



```
copy_from_user(buffer, p, count);
set_up_DMA_controller( );
scheduler( );
```

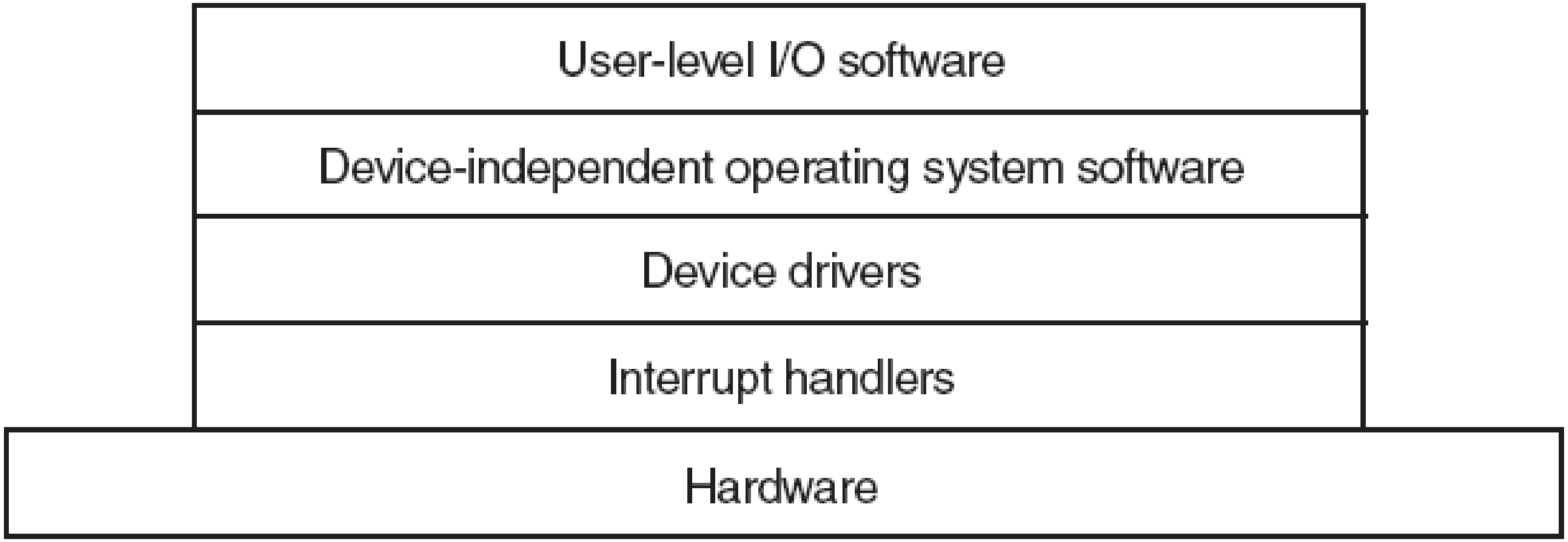
(a)

```
acknowledge_interrupt( );
unblock_user( );
return_from_interrupt( );
```

(b)

Figure 5-10. Printing a string using DMA. (a) Code executed when the print system call is made. (b) Interrupt service procedure.

I/O Software Layers



User-level I/O software

Device-independent operating system software

Device drivers

Interrupt handlers

Hardware

Figure 5-11. Layers of the I/O software system.

Device Drivers

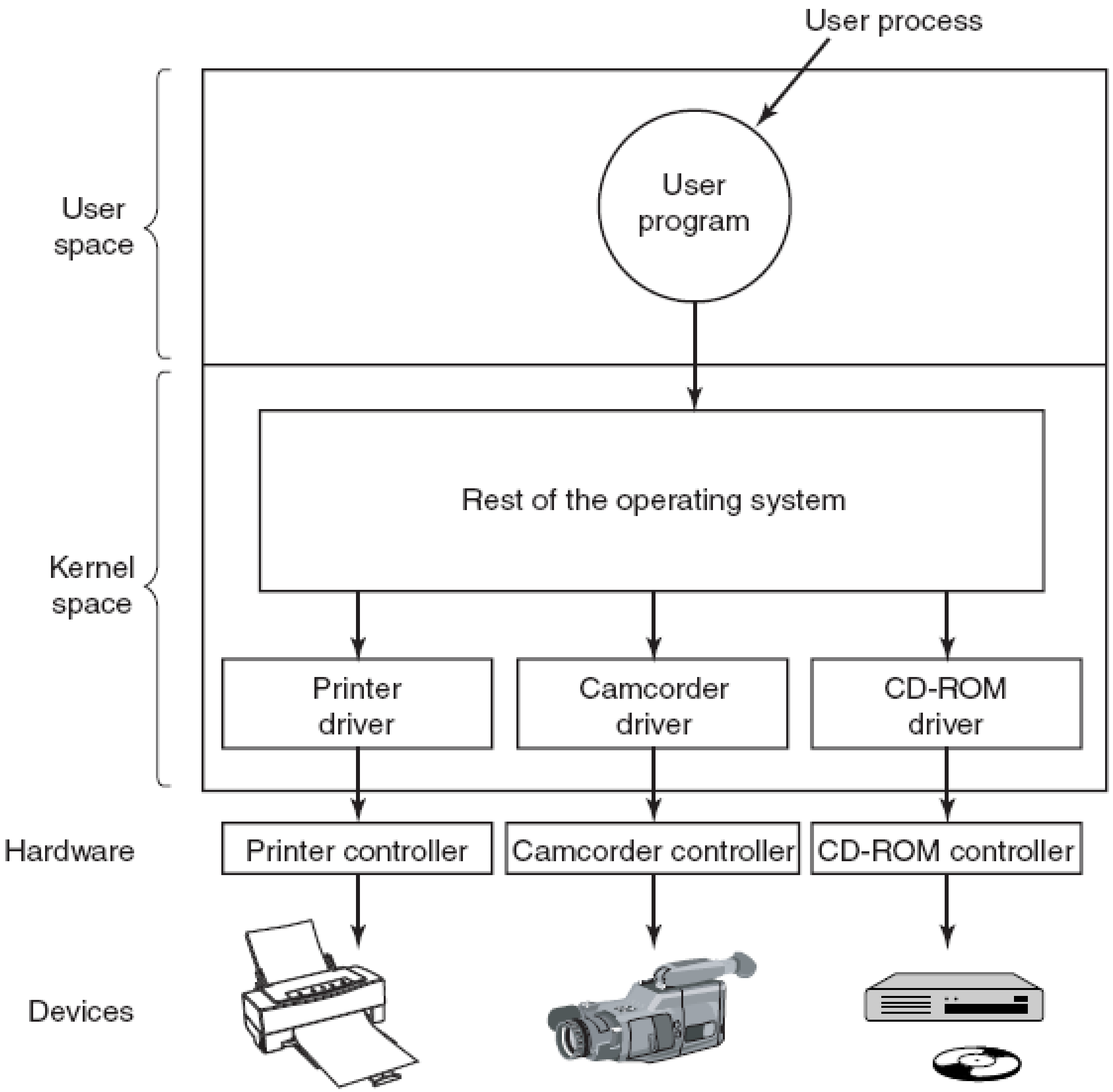
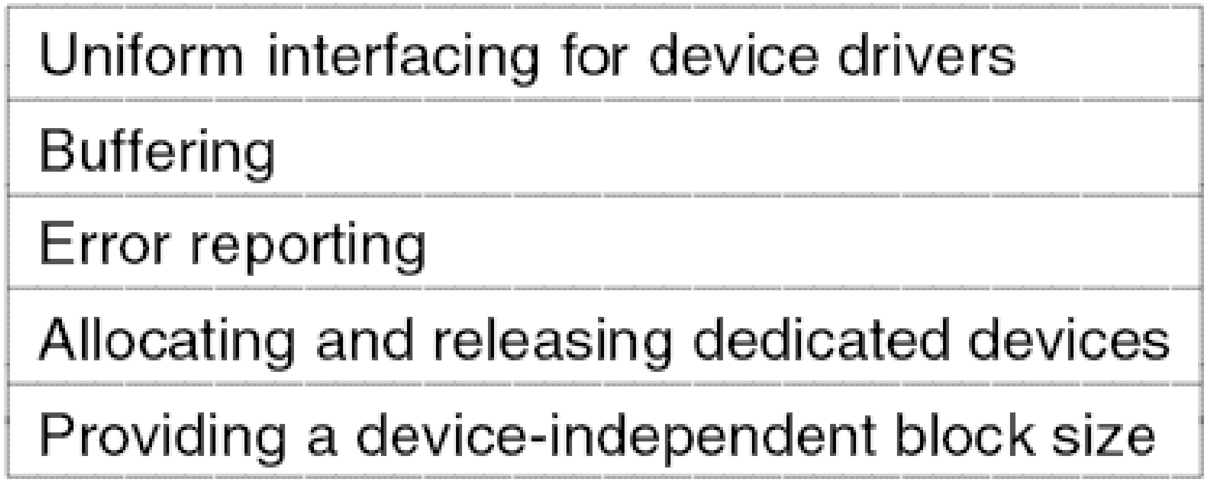


Figure 5-12. Logical positioning of device drivers. In reality all communication between drivers and device controllers goes over the bus.

Device-Independent I/O Software



Uniform interfacing for device drivers
Buffering
Error reporting
Allocating and releasing dedicated devices
Providing a device-independent block size

Figure 5-13. Functions of the device-independent I/O software.

Uniform Interfacing for Device Drivers

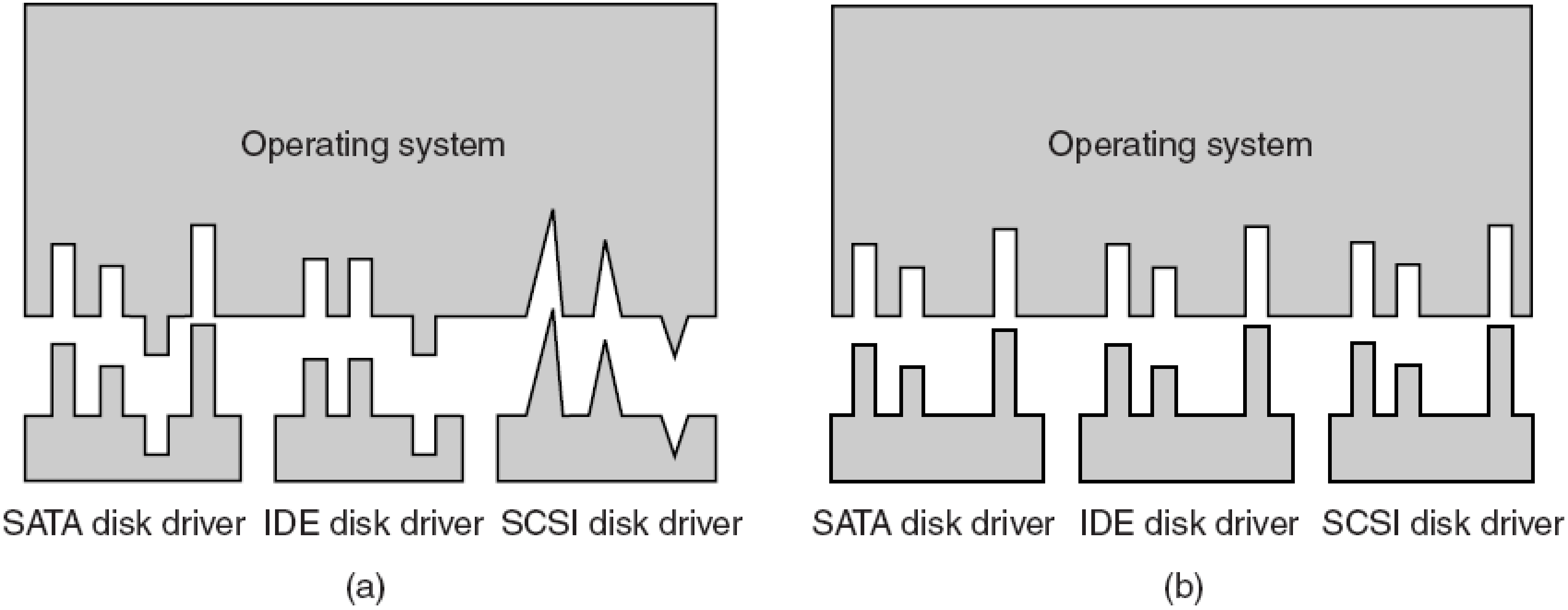


Figure 5-14. (a) Without a standard driver interface.
(b) With a standard driver interface.

Buffering

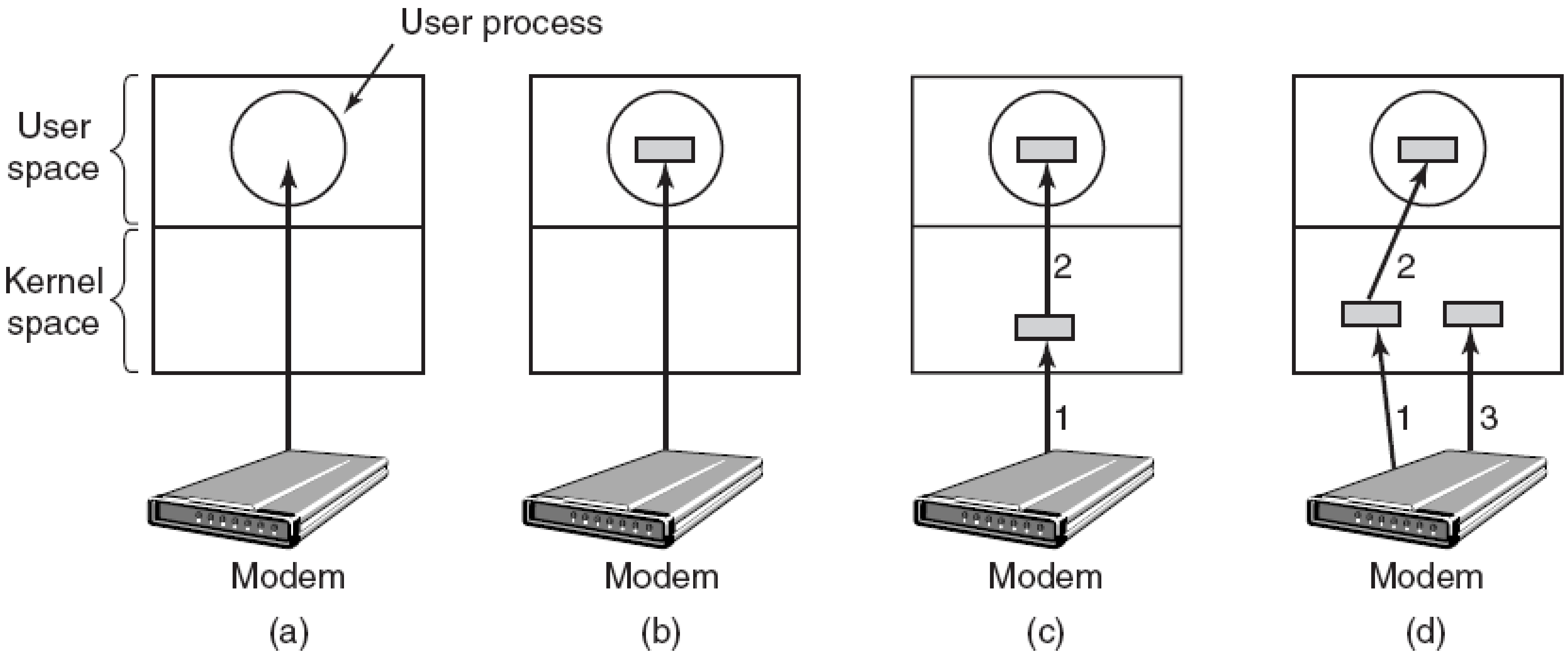
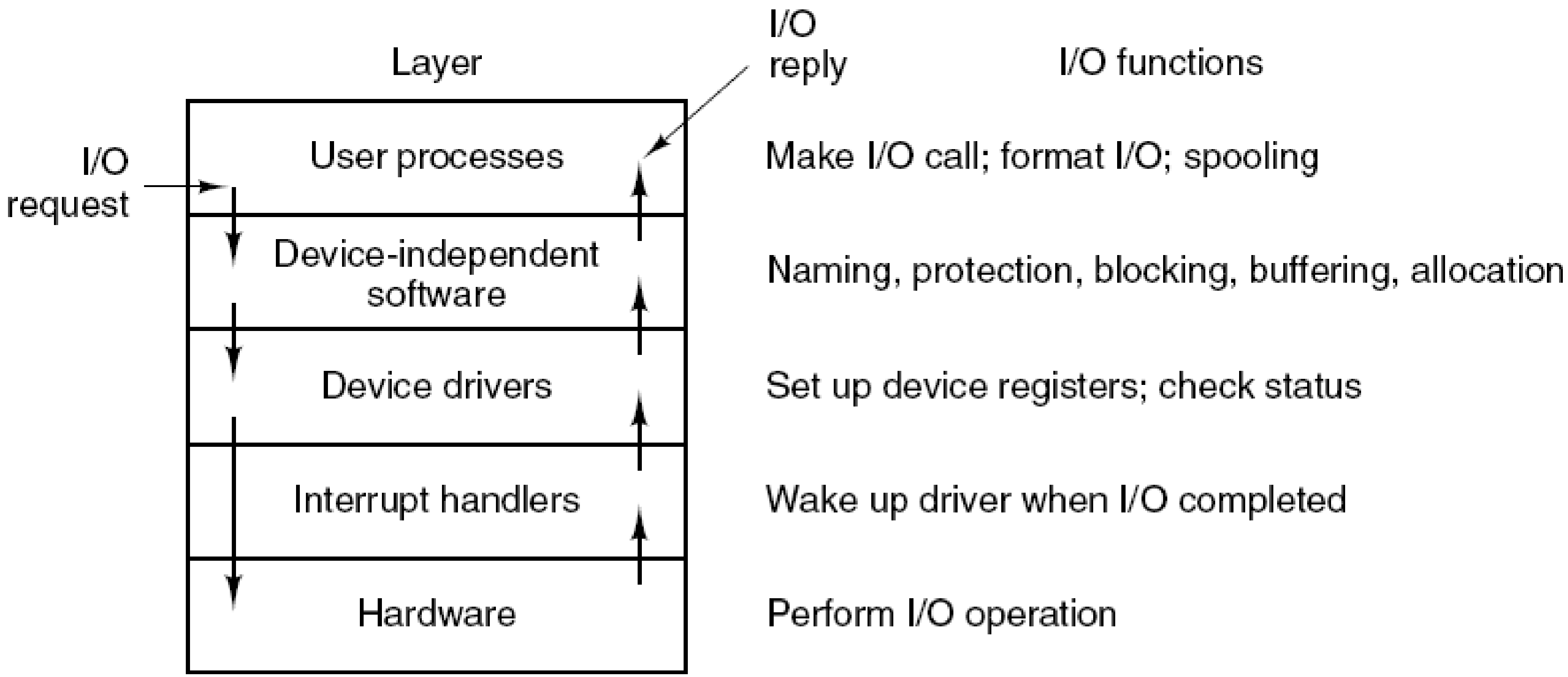


Figure 5-15. (a) Unbuffered input. (b) Buffering in user space. (c) Buffering in the kernel followed by copying to user space. (d) Double buffering in the kernel.

User-Space I/O Software



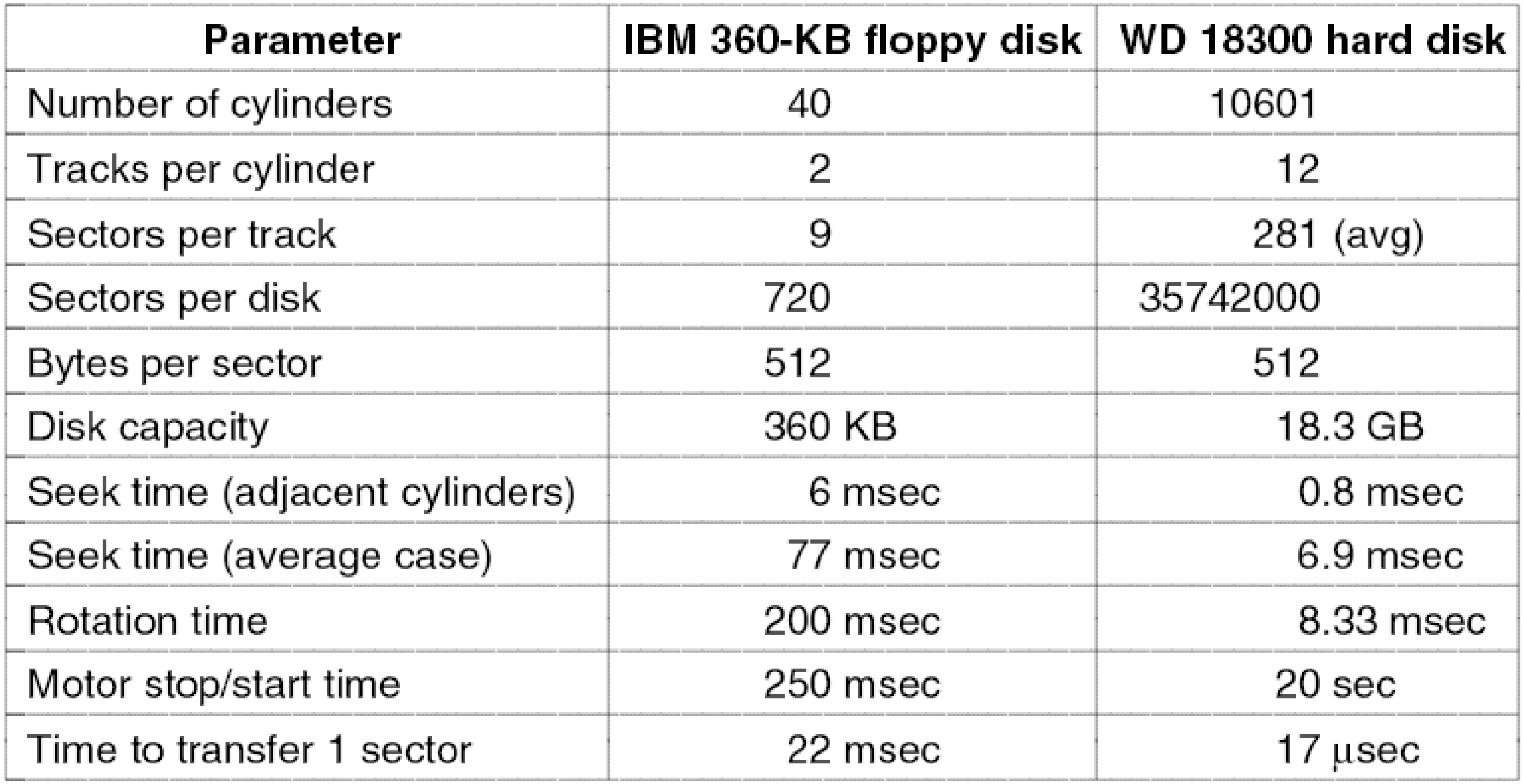
I/O request

I/O reply

Layer	I/O functions
User processes	Make I/O call; format I/O; spooling
Device-independent software	Naming, protection, blocking, buffering, allocation
Device drivers	Set up device registers; check status
Interrupt handlers	Wake up driver when I/O completed
Hardware	Perform I/O operation

Figure 5-17. Layers of the I/O system and the main functions of each layer.

Magnetic Disks



Parameter	IBM 360-KB floppy disk	WD 18300 hard disk
Number of cylinders	40	10601
Tracks per cylinder	2	12
Sectors per track	9	281 (avg)
Sectors per disk	720	35742000
Bytes per sector	512	512
Disk capacity	360 KB	18.3 GB
Seek time (adjacent cylinders)	6 msec	0.8 msec
Seek time (average case)	77 msec	6.9 msec
Rotation time	200 msec	8.33 msec
Motor stop/start time	250 msec	20 sec
Time to transfer 1 sector	22 msec	17 μsec

Figure 5-18. Disk parameters for the original IBM PC 360-KB floppy disk and a Western Digital WD 18300 hard disk.

RAID (1)

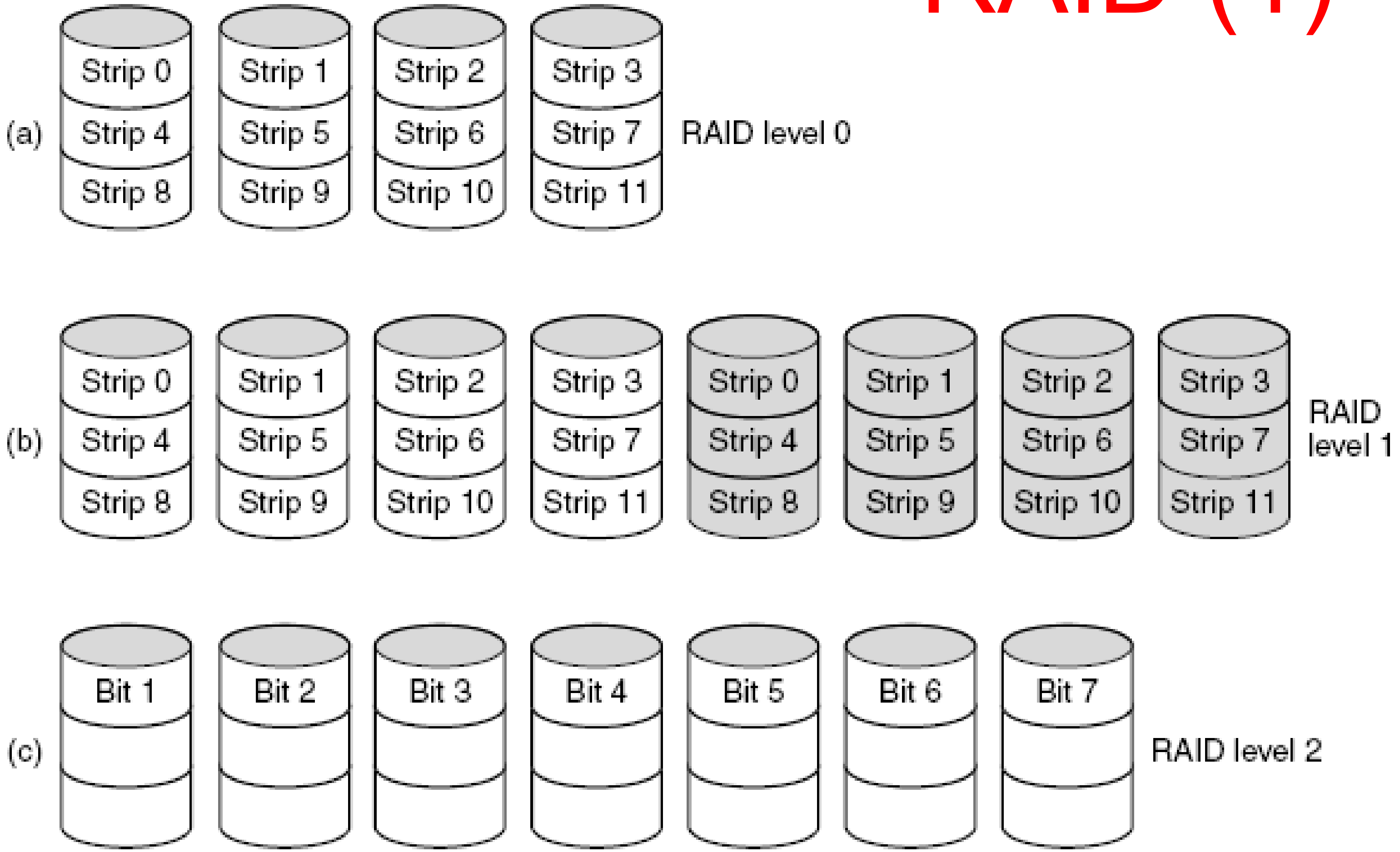


Figure 5-20. RAID levels 0 through 5.
Backup and parity drives are shown shaded.

RAID (2)

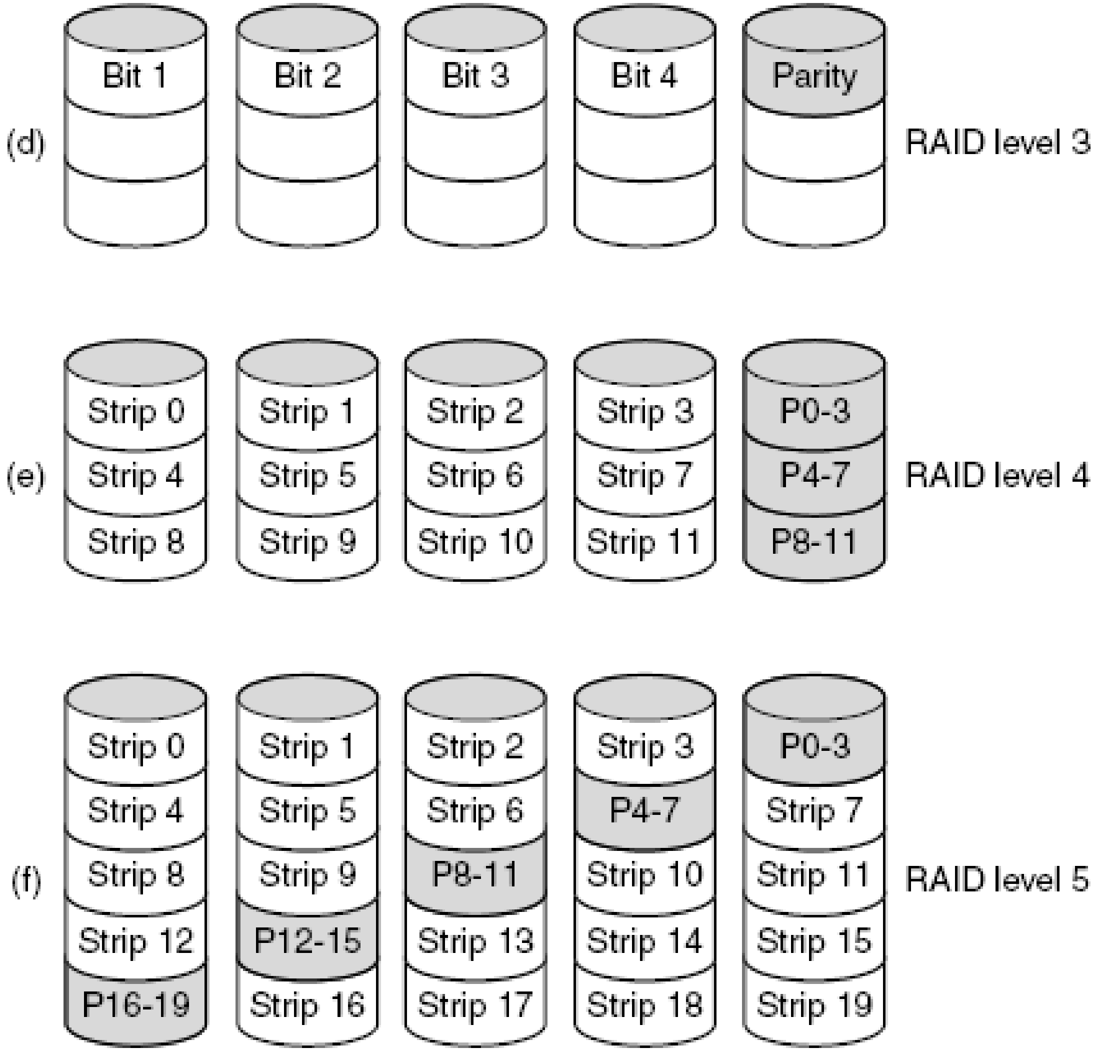


Figure 5-20. RAID levels 0 through 5.
Backup and parity drives are shown shaded.

CD-ROMs

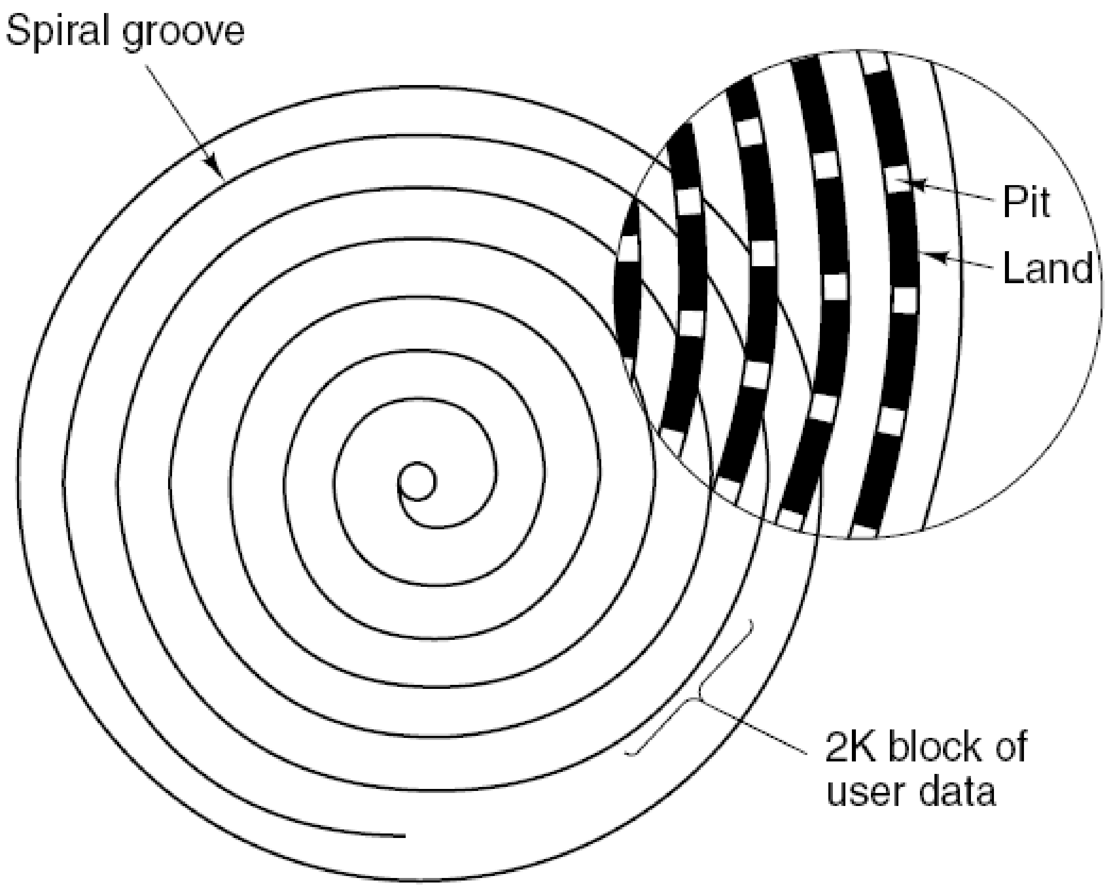


Figure 5-21. Recording structure of a compact disc or CD-ROM.

CD-Recordables

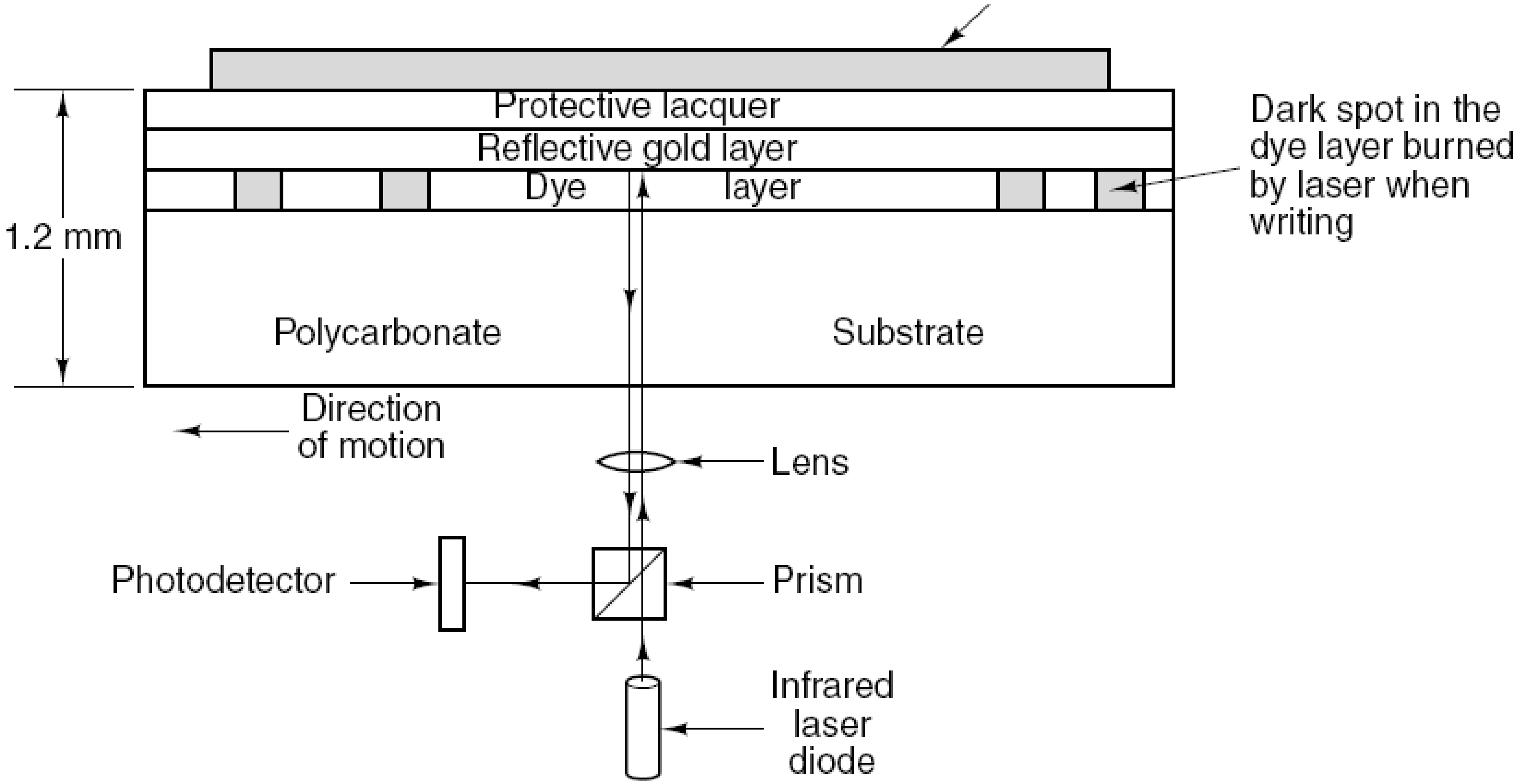


Figure 5-23. Cross section of a CD-R disk and laser. A silver CD-ROM has similar structure, except without dye layer and with pitted aluminum layer instead of gold layer.

DVD

DVD Improvements on CDs

1. Smaller pits
   (0.4 microns versus 0.8 microns for CDs).
2. A tighter spiral
   (0.74 microns between tracks versus 1.6 microns for CDs).
3. A red laser
   (at 0.65 microns versus 0.78 microns for CDs).

DVD (2)

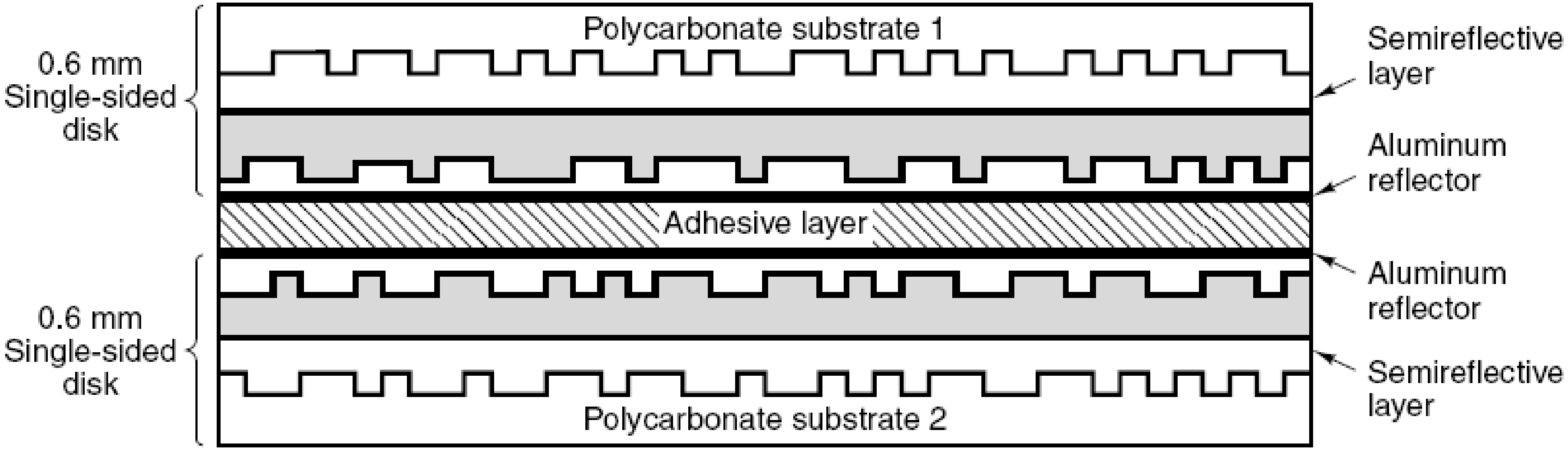
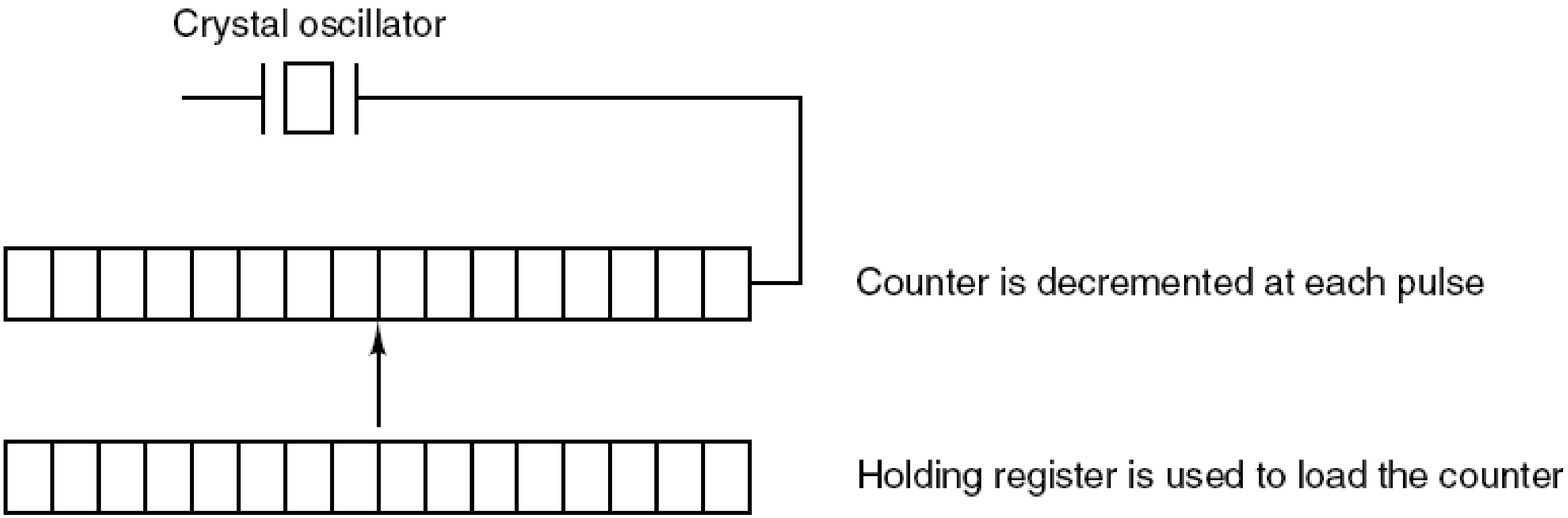


Figure 5-24. A double-sided, dual-layer DVD disk.

Clock Hardware



Crystal oscillator

Counter is decremented at each pulse

Holding register is used to load the counter

Figure 5-32. A programmable clock.

Clock Software

Typical duties of a clock driver

1. Maintaining the time of day.
2. Preventing processes from running longer than they are allowed to.
3. Accounting for CPU usage.
4. Handling alarm system call made by user processes.
5. Providing watchdog timers for parts of the system itself.
6. Doing profiling, monitoring, statistics gathering.

Soft Timers

Soft timers succeed according to rate at which kernel entries are made because of:

1. System calls.
2. TLB misses.
3. Page faults.
4. I/O interrupts.
5. The CPU going idle.

The X Window System (1)

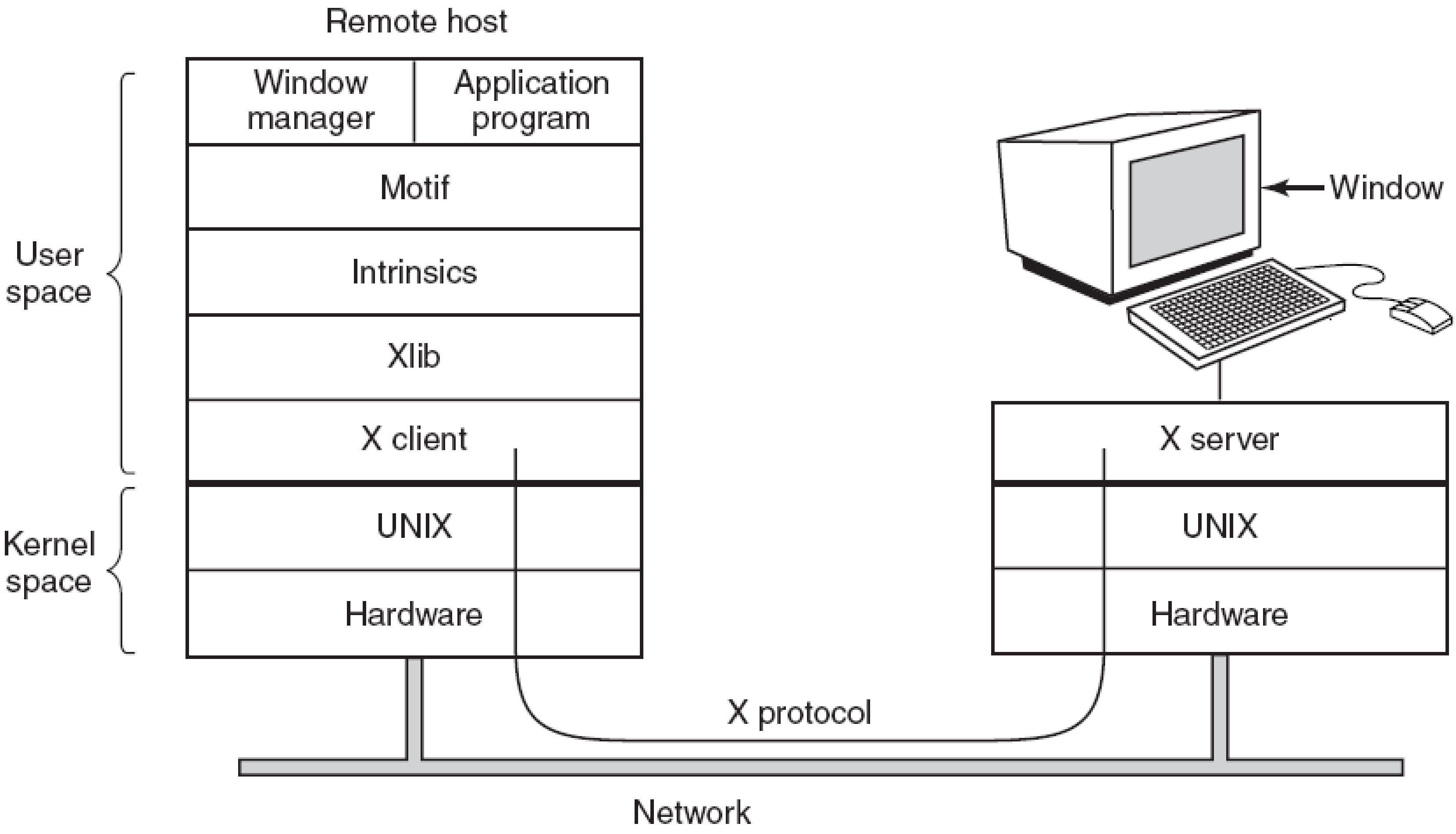


Figure 5-37. Clients and servers in the M.I.T. X Window System.

The X Window System (2)

Types of messages between client and server:

1. Drawing commands from the program to the workstation.
2. Replies by the workstation to program queries.
3. Keyboard, mouse, and other event announcements.
4. Error messages.