

FIXED DISKS AND THE PC AT

*A comparison of speed and performance
that matches the PC AT against the PC XT*



BY JON SHIELL AND JOHN MARKOFF

THE IBM PC AT provides more flexibility and computing power than its popular predecessor, the IBM PC. The AT is based on the Intel 80286 microprocessor, which executes instructions faster and more efficiently than the PC's 8088.

The AT provides a memory space potentially 16 times larger than the PC's. This is an unexplored realm for personal computer software developers that is certain to prove tempting.

The introduction of the AT set a new standard for fixed-disk capacity and performance for microcomputer systems. The AT provides double the disk capacity of the PC XT and offers a simple method for the user to install up to two hard disks with capacities of almost 140 megabytes each.

WHY THE AT IS FASTER THAN THE XT

There are a number of factors, some obvious, some more subtle, why the AT exceeds the XT's performance. First, the raw clock rate of the AT's 80286 chip is 12 MHz. (The 80286 divides this by 2 internally.) The XT's 8088 clock rate is 4.77 MHz. Therefore, even if everything else were comparable between the two systems, the AT would still be a little more than 26 percent faster than the PC.

However, everything is not that



clearly defined. For example, the AT's math coprocessor, the 80287, runs at only one-third the frequency of the AT's raw system microprocessor clock (12 MHz)—that is, 4 MHz. This compares unfavorably to the IBM PC's numeric processor, the 8087, which runs at the full speed of the PC—4.77 MHz. The math coprocessor doesn't help the AT's performance as much as might be expected.

Despite this obvious imbalance, however, most comparisons lean markedly in favor of the AT. A comparison of each computer's system bus reveals that the AT has a 16-bit data bus; the PC has an 8-bit data bus. Using an 8-bit bus for a 16-bit microprocessor (as is done in the IBM PC) costs 15 to 20 percent in system performance. This, combined with the approximately 25 percent increase due to the difference in clock rates, still gives a speedup of less than 50 percent, so there must still be other factors that contribute to the AT's speed advantage.

Significantly, the 80286-powered AT runs with a single wait state; if it ran no wait states, RAM random-access read/write memory) with a maximum access time of less than 120 nanoseconds (ns) would be necessary. By adding the wait state it is possible for the AT to use standard RAM (150-ns access). The use of the wait state cuts the effective processing rate by less than 25 percent.

(continued)

Jon Shiell is a system architect and microprogrammer. He can be reached at POB 61195, Sunnyvale, CA 94088.

John Markoff, a technology writer for the San Francisco Examiner, is a former BYTE editor. He can be reached at 110 Fifth St., San Francisco, CA 94123.

cent on the average, according to Intel benchmarks.

At the microprocessor level, the 80286 is dramatically more efficient than the 8086. The 8086/8088 merely overlaps the instruction fetch (IF) of the next instruction with the execution of the current one.

The 80286 architecture is "pipelined" and adds an additional level of parallelism by overlapping the fetch of a third instruction with the decod-

ing of a second instruction and execution of the current instruction. In fact, the amount of instruction overlap in the 80286 is roughly comparable to that in some of IBM's mainframe computers.

When combined with the additional hardware speedup of some instructions and internal bus and clock speed enhancements, the 80286 runs between two and six times faster than a 5-MHz 8086 or the PC XT's

4.77-MHz 8088, which is slower and uses an 8-bit data bus.

Another interesting point of comparison is in the speed of DMA (direct memory access), the device-to-memory transfer that doesn't involve the central processing unit (CPU). Here, the AT's performance is significantly slower than that of the IBM PC.

The PC AT's system DMA rate is 1.66 microseconds (μ s) per transfer

FACTORS AFFECTING DISK PERFORMANCE

Four major physical factors determine overall disk performance: access time, cylinder size, transfer rate, and average latency.

Access time is the amount of time it takes to move the read/write heads over the desired tracks (cylinders). Once the heads are over the desired tracks, they must settle down from the moving height to the read/write height. This is called the settling time and is normally included in the access time. Specifications for AT and XT disk-drive options are shown in table A.

A cylinder is composed of all tracks that are under the read/write heads at one time. Thus, tracks per cylinder is the same as the number of data heads in the drive. Cylinder size is defined as tracks/cylinder \times sectors/track \times bytes/sector.

The Quantum Q540, for example, has four platters and eight data heads,

while the Vertex V170 has four platters, seven data heads, and one servo head. The difference is that the Quantum drive uses an embedded (or wedge) servo, where the servo signal is embedded on the data tracks, preceding the data portion of each sector on the disk. The Vertex drive uses a dedicated servo that requires its own surface. This difference means that the Quantum drive has 8.5K bytes more data available to it before it must seek the next track; if all other factors were equal (which they aren't), the Quantum would be slightly faster in those cases that required reading that "extra" 8.5K bytes.

Transfer rate is the rate at which data comes off the disk. It depends on rotation rate, bit density, and sector interleaving. The first two factors are practically the same for all AT-compatible 5 $\frac{1}{4}$ -inch hard disks, but not for all

floppy disks (the AT's spin 20 percent faster than the other PC floppies).

Sector interleaving is used to cut down the effective transfer rate. The interleave factor of 6 used on the XT cuts the effective transfer rate from 5 megabits per second to 0.833 megabit per second. Note that embedded servo disks, such as those used in the XT and the AT, actually spin about 1 percent slower than 3600 revolutions per minute (rpm) to allow for the increased density due to the servo.

Average latency is the time required for a disk to spin one-half of a revolution. For hard disks, which spin at 3600 rpm, the average latency is 8.33 ms (1/3600 rpm \times 60 seconds/minute \times 0.5 = 8.33 ms per half revolution). This is due to the fact that after the heads finish seeking and settling, you must wait for the required sector to come under the heads.

Table A: Sample transfer rates and a comparison of specifications for the disk-drive options available for the IBM PC AT and PC XT. Note that the interleave factor is the only difference between the last two entries in the table; the drive is the same for both entries.

	PC XT Floppy Disk	AT 360K-byte Floppy Disk	AT 1.2-megabyte Floppy Disk	PC XT Hard Disk	PC AT Hard Disk	PC AT Hard Disk
Rotation	300 rpm	360 rpm	360 rpm	3600 rpm	3600 rpm	3600 rpm
Base rate	250K bits/second	300K bits/second	500K bits/second	5M bits/second	5M bits/second	5M bits/second
Interleave	none	none	none	6	3*	2*
Actual rate in bytes/second	31,250	37,500	62,500	104,167	208,333	312,500

* The AT technical reference manual shows an interleave factor of 2, but the advanced diagnostics and the "IBM PC Seminar Proceedings" claim that the interleave factor is 3.

(five AT cycles per transfer at 3 MHz). This yields the channel bandwidths shown in table 1.

The DMA rate of the PC XT is 1 byte per five cycles (1.05 μ s) for a DMA channel bandwidth of 0.952 megabyte per second. This bandwidth is the result of IBM designing the AT with 5-MHz DMA controllers.

The AT's DMA controllers cannot run at the full 6-MHz rate of the AT system clock, and it is difficult to attain clock rates that are not simply an integer division of the system timing. The design decision was to set the DMA rate at 3 MHz. This is not as critical a shortcoming for the AT as it may first appear, because few devices (hard disks excluded) can saturate the AT's DMA capability.

It is interesting that the AT does not use DMA for hard-disk input/output (I/O). Instead, the AT's hard-disk controller has a 512-byte sector buffer that is accessible by the 80286 as a 16-bit device.

When the buffer is full or empty, the controller interrupts the 80286 (using INT14 hexadecimal), which then moves the data via programmed I/O (that is, Rep Insw for reads and Rep Outsw for writes) to or from memory at a rate of 2 megabytes per second. This transfer rate is approximately twice that of the XT.

The 2-megabyte transfer rate of the AT for string-move operations is the result of using three cycles (including one wait state) to bring the data (16 bits) into the processor and another three cycles (including another wait state) to move the data into memory. This process uses six cycles per 2-byte transfer. At a clock rate of 6 MHz, a single clock cycle takes 167 ns. Six cycles therefore require 1002 ns for a 2-byte transfer, which corresponds to an effective rate of 2 megabytes per second.

ZERO WAIT STATE

IBM has provided a special line called the zero wait state (OWS) signal on the bus to suppress the wait states. This is useful if you have high-speed memory and an expansion card that supports this signal.

For example, in the preceding case performance could be improved by one-sixth because the memory card used in the transfer could suppress the wait state. However, the AT's disk controller won't suppress the wait state when transferring data from its sector buffer. Therefore, a savings of only one cycle per transfer is possible. By using the OWS signal we can speed up the transfer rate from 2 megabytes per second to 2.5 megabytes per second.

In addition to bus factors, the AT's 1.2-megabyte floppy disk holds more than three times as much data as the XT's 360K-byte floppy disk. It also has a data-transfer rate that is twice that of the XT's and an average latency that is 20 percent less than that of the XT's. Furthermore, the AT's floppy disk has somewhat faster access times.

Of course floppy-disk speeds are not that significant, as the floppy disk plays a much less important role in *(continued)*

Table 1: Channel bandwidth data for the IBM PC AT.

DMA Channel	Transfer Width	Channel Bandwidth
0-3	1 byte	0.6 megabyte/second
4	used to cascade channels 0-3	
5-7	2 bytes	1.2 megabytes/second

CAN THE AT SUPPORT NONINTERLEAVED HARD DISKS?

[Editor's note: An interleave factor of 1:1 implies a noninterleaved disk.]

Let's do some quick back-of-the-napkin calculations to answer this question. At 3600 rpm one rotation takes 16.67 ms, and one rotation per track of 17 sectors results in less than 1 ms per sector. There are about 10,416 bytes per unformatted track, again at one track of 17 sectors; this gives us about 612 bytes per unformatted sector.

There are 512 bytes of data in a formatted sector, thus about one-sixth of the sector is not data (512 data bytes out of 612 possible). If we assume that during this period the AT's track buffer is available to the 80286, then we have one-sixth of 1 millisecond or about 150 μ s to fill or empty the buffer. This assumes that the buffer is single-ported and doesn't support interleaved access, which seems to be the case for the AT.

Programmed I/O normally transfers one word every six cycles, which means a data rate of 2 megabytes per second. To move our 512 bytes of data will take about 250 μ s, assuming INT14 hexadecimal (the buffer full/empty interrupt) overhead is 0, which it isn't.

We expect that without interleaving we have less than 150 μ s to fill or empty the buffer, while the 80286 would need at least 250 μ s. Consequently, the sector buffer will not support 1:1 interleaving but will probably support 2:1 interleaving without any major problems. This is due to the fact that we have almost 1 millisecond to fill or empty the buffer when 2:1 interleaving is used.

the system design of the AT than it does in the PC.

COMPARING THE AT AND XT HARD DISKS

Let's compare the hard disks that are standard equipment on the XT and

the AT. We will also discuss a 140-megabyte disk, the Maxtor XT-1140.

Looking at the relative access times (including latency) for the XT, the AT, and the Maxtor shows the Maxtor to be by far the fastest drive; however, we also find that the AT's disk is about

40 percent faster on the average than the XT's (see table 2).

But access time is not everything. Table 3 shows the comparative times for a 127,000-byte read (assume all sectors are contiguous and start with sector 1, head 0).

Although the Maxtor's access time for this operation is about 350 percent faster than the AT's, which is about 15 percent faster than the XT's, its access time gets swamped by the transfer times. The apparent differences are shown in table 4.

The XT appears to be about 85 percent slower than the AT, which is only about 14 percent slower than the Maxtor. The Maxtor's faster access time is swamped by the 609 milliseconds (ms) it takes to transfer the data, and it has no advantage in transfer rates. However, for a smaller (more normal size) file, say 34K bytes, we would get the times shown in table 5.

The XT appears to be about 80 percent slower than the AT, which is

Table 2: A comparison of disk performance for the AT and the XT with standard disks and the AT with the Maxtor disk.

Track-to-track access time + latency		Index
Maxtor	5 ms + 8.3 ms = 13.3 / 22.3	= 0.60
PC AT	14 ms + 8.3 ms = 22.3 / 22.3	= 1.00
PC XT	16 ms + 8.3 ms = 24.3 / 22.3	= 1.09
Mean access time + latency		Index
Maxtor	30 ms + 8.3 ms = 38.3 / 60.3	= 0.64
PC AT	52 ms + 8.3 ms = 60.3 / 60.3	= 1.00
PC XT	85 ms + 8.3 ms = 93.3 / 60.3	= 1.55
Maximum access time + latency		Index
Maxtor	48 ms + 8.3 ms = 56.3 / 105.3	= 0.53
PC AT	97 ms + 8.3 ms = 105.3 / 105.3	= 1.00
PC XT	205 ms + 8.3 ms = 213.3 / 105.3	= 2.03

Table 3: Time required for a 127,000-byte read operation.

Drive Type	Bytes/Cylinder	Cylinders Required	Average Access Time (note 1)	Cylinder-to-Cylinder Access Time (note 2)	Interleave Factor	Average Latency/Track (note 3)	Aggregate Access Time (note 4)	Transfer Rate (bytes/second)	Transfer Time (note 5)
PC XT	34K	four	85	(3) x 16	6	8.3	166.2	104,200	1219
PC AT	34K	four	65	(3) x 14	3	8.3	140.2	208,333	609
Maxtor	127K	one	30	n.a. (one cylinder used)	3	8.3	38.3	208,333	609

Notes

1. Average access time is the average time required for the disk heads to move from any cylinder to any other cylinder on the disk (including settling time before the heads can read data).
2. Cylinder-to-cylinder access time is the time required to move the heads from one cylinder on the disk to an adjacent cylinder on the disk (including settling time).
3. Average latency is the time required for a disk to spin one-half revolution. For hard disks, which spin at 3600 rpm, the average latency is 8.33 milliseconds
(1/3600 rpm x 60 seconds/minute x 0.5 = 8.33 ms per half revolution).
This is due to the fact that after the heads finish seeking and settling, you must wait for the required sector to come under the heads before reading data.
4. Aggregate access time is determined by
average access + [number of cylinder-to-cylinder changes x cylinder-to-cylinder time] + [number of cylinders x average latency].
5. Transfer time is determined by
[127K bytes divided by transfer rate in bytes/second] expressed in milliseconds.

about 40 percent slower than the Maxtor. If the file were spread over two cylinders on the XT and the AT (a more normal case due to their significantly smaller cylinder size), the ratios would be as shown in table 6. As expected, the Maxtor looks even better in this more typical case because it doesn't need the additional track-to-track access and associated latency. Also note that the AT's relative performance slipped a bit due to the small difference in AT and XT track-to-track access times.

Access time is most important when doing a lot of disk accesses in a short period of time, for instance, when searching a large database or working on a file that, due to fragmentation, is spread out over the entire disk. For example, to read a 34K-byte file (17 clusters on the AT's 20-megabyte disk) on a very fragmented disk could take up to 17 average accesses for a total of more than 1 second (1025 ms = 17 × (52 ms + 8.3 ms)).

It takes 163 ms to transfer the data. So our effective I/O time is 1188 ms as opposed to 236 ms for an unfragmented disk. Even if we assume less fragmentation—say the file is split into four parts—we get total access time of about 241 ms (4 × (52 ms + 8.3 ms)); adding the transfer time gives us 404 ms, or about a 50 percent decrease in the disk's effective "speed."

This points out the importance of not letting your disk get very fragmented, as this results in much longer effective access times.

Let's look at another example using our big file of 127K bytes. If it's all in three continuous cylinders, it will take about 117.9 ms, as we've shown, for the access portion of the read (assuming that the file allocation table [FAT] and the directory are already in memory). But the worst case is where the file is spread out over the entire disk, with only one cluster per cylinder and the cylinders in random order.

It could take approximately one mean access per track over 64 tracks (127K bytes divided into 64 clusters of 2K bytes each; one cluster per track) for an effective access time of

Table 4: A comparison of access time, transfer time, and performance index for the AT and the XT with standard disks and the AT with the Maxtor disk.

Drive Type	Access Time	Transfer Time	Total	Performance Index
PC XT	166.2 ms	+ 1219 ms	= 1385 ms / 749 ms	= 1.85
PC AT	140.2 ms	+ 609 ms	= 749 ms / 749 ms	= 1.00
Maxtor	38.3 ms	+ 609 ms	= 647 ms / 749 ms	= .86

Table 5: Transfer time and performance index for a 34K-byte transfer.

PC XT drive: one average access			
	85 ms	+ 8.3 ms	= 93.3 ms
	34,000 bytes at 104,200 bytes/second		= 326 ms (interleave of 6)
PC AT drive: one average access			
	65 ms	+ 8.3 ms	= 73.3 ms
	34,000 bytes at 208,333 bytes/second		= 163 ms (interleave of 3)
Maxtor drive: one average access			
	30 ms	+ 8.3 ms	= 38.3 ms
	34,000 bytes at 208,333 bytes/second		= 163 ms (interleave of 3)

Results in:

Drive Type	Access Time	Transfer Time	Total	Performance Index
PC XT	93.3 ms	+ 326 ms	= 419 ms / 236 ms	= 1.78
PC AT	73.3 ms	+ 163 ms	= 236 ms / 236 ms	= 1.00
Maxtor	38.3 ms	+ 163 ms	= 146 ms / 236 ms	= .62

Table 6: Transfer time and performance index for a 34K-byte file contained on two cylinders.

Drive Type	Access Time	Transfer Time*	Total	Performance Index
PC XT	93.3 ms	+ 326 ms	+ 24.3 ms	= 444 ms / 259 ms = 1.71
PC AT	73.3 ms	+ 163 ms	+ 22.3 ms	= 259 ms / 259 ms = 1.00
Maxtor	38.3 ms	+ 163 ms	+ 0.0 ms	= 146 ms / 259 ms = .56

* Track-to-track access time plus average latency

almost 4 seconds (3859 ms = 64 × 60.3 ms including latency). So we get disk I/O times of about 749 ms versus 4468 ms (almost 4.5 seconds); thus, the fragmented disk appears to be about six times slower than the unfragmented one.

If you want to get an idea of how fragmented your hard disk is, the DOS CHKDSK command, when given a filename, displays the number of

discontinuous areas occupied by the file. While this is not as useful as it could be (what we really want to know is how many cylinders the file is spread across), it gives us a fair idea of the amount of fragmentation.

CLUSTER SIZE

An additional factor, cluster size, is purely an operating-system function.

(continued)

but it does affect disk performance. A cluster is the number of contiguous sectors that DOS allocates each time disk space is needed. The FAT is what DOS uses to keep track of which clusters are allocated to which files and which are available for use. Beginning with DOS 3.0 there are now two types of FATs, depending on the size of the partition. An FAT that uses 12-bit entries is used for disks holding

less than 20 megabytes, and one with 16-bit entries is used for disks holding 20 megabytes or more.

On a 10-megabyte disk or partition, each cluster is 4K bytes or eight sectors, and the FAT takes up 4K bytes. Using a 12-bit FAT entry results in 4096 possible clusters. DOS uses a 4K-byte cluster with a 12-bit FAT. This results in a maximum disk size of 16 megabytes (4096 clusters/disk × 4096

bytes/cluster) without using a larger cluster size. All floppy disks use this scheme.

On disks or partitions of 20 megabytes and more, the cluster size is 2K bytes or four sectors. For a 20-megabyte disk, this means that the FAT is 20K bytes, and for a 32-megabyte disk or partition, the FAT occupies 32K bytes.

(continued)

USING DEBUG TO DETERMINE THE CLUSTER SIZE OF A DISK

The first and last register dumps are for the C drive, while dump 2 is a 1.2-megabyte floppy disk.

The register meanings are as follows: If the AX register contains FFFF hexadecimal on return, then the drive number was invalid and the rest of the registers are meaningless. Otherwise the registers contain the following:

AX number of sectors per cluster, four for the hard disk and one for the floppy disk

BX number of free clusters

CX number of bytes per sector, normally 200 hexadecimal or 512 bytes

DX total number of clusters on the disk

Therefore, the hard disk has a DOS partition of 15,630 (3D0E hexadecimal) clusters at four sectors per cluster, for a total of 62,520 sectors or 31.26 megabytes (using a Quantum Q540, Type 7 drive).

```
>debug
-a
XXXX:0100 ; Remember that debug works only with hexadecimal numbers
XXXX:0100 mov ax,3600 ; Load AH with the function code (36 hexadecimal)
XXXX:0103 mov dx,0000 ; DL = drive number (0= default, 1=A, 2=B, 3=C, ...)
XXXX:0106 int 21 ; Call DOS to do the function
XXXX:0108 nop ; Space saver to stop at
XXXX:0109
-g = 100 108 ; This is for Drive C

AX=0004 BX=2F6C CX=0200 DX=3D0E SP=XXXX BP=XXXX SI=XXXX DI=XXXX
DS=XXXX ES=XXXX SS=XXXX CS=XXXX IP=0108 XX XX XX XX XX XX XX XX
XXXX:0108 90 NOP
-a 103
XXXX:0103 mov dx,0001 ; Now let's look at the 1.2-megabyte floppy
XXXX:0106
-g = 100 108

AX=0001 BX=045B CX=0200 DX=0943 SP=XXXX BP=XXXX SI=XXXX DI=XXXX
DS=XXXX ES=XXXX SS=XXXX CS=XXXX IP=0108 XX XX XX XX XX XX XX XX
XXXX:0108 90 NOP
-a 103
XXXX:0103 mov dx,0003 ; Now let's look at Drive C
XXXX:0106
-g = 100 108

AX=0004 BX=2F6C CX=0200 DX=3D0E SP=XXXX BP=XXXX SI=XXXX DI=XXXX
DS=XXXX ES=XXXX SS=XXXX CS=XXXX IP=0108 XX XX XX XX XX XX XX XX
XXXX:0108 90 NOP
-q
>
```


Statistical Tools for PROFESSIONALS

- ▶ easy to learn
- ▶ fast
- ▶ comprehensive
- ▶ easy to use
- ▶ accurate
- ▶ reliable

StatPac™

Complete statistical package proven for over 5 years



Goodness-of-Fit™

Advanced regression package for interactive model building

Forecast Plus™

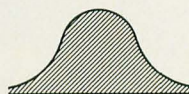
Over a dozen graphical time-series analysis techniques

Call for Information 1-800-328-4907

WALONICK ASSOCIATES

6500 Nicollet Ave. S., Minneapolis, MN 55423

(612) 866-9022



The actual calculation for FAT size, assuming a 2K-byte cluster, is: disk or partition size/2K bytes × 2 bytes/FAT entry, which reduces to 1K byte of FAT per 1 megabyte of disk or partition. The reason for increasing the number of clusters on the disk instead of just increasing the size of each cluster is that, while large clusters are good for access time, most files fit in just a few clusters, thereby minimizing the fragmenting of the disk.

A large cluster is inefficient in terms of space. Assume that we use an 8K-byte (16 sectors) cluster size for our 20-megabyte disk. This lets us map the disk into an FAT with 12-bit entries (20 megabytes at 1 cluster; 8K bytes is 2560 clusters). But because files are allocated space in units of 8K bytes, on the average each file wastes 4K bytes (half of its last cluster).

This 4K bytes of waste per file can really add up. Let's say we have 512 files on a disk. This means that 2 megabytes, or 10 percent of the total space on the disk, is wasted (512 files × 4K bytes per file). By keeping the cluster size at 2K bytes (four sectors), the amount of wasted space is greatly reduced, but a penalty is paid both in the size of the FAT and in the speed with which the disk gets fragmented.

The reason we care about the size of the FAT is not because it uses up more disk space but because DOS keeps a copy of the FAT of the active disk(s) in memory. And while 20K bytes for a 20-megabyte disk (or 32K bytes for the biggest disk or partition DOS supports) is not so bad in a 256K-byte system, it can add up.

One last relevant feature of the AT is that its hard-disk controller supports overlapped (buffered) seeks. Also, the controller board supports one floppy-disk and one hard-disk data transfer at the same time. The fixed-disk interface is ST412. Overlapped seeks allow a system with two or more hard disks to overlap operations. This is done by sending multiple step (seek) pulses to a drive that then disconnects and does the stepping without the controller. The controller is then free to work with its remaining drive(s). ■

For users of IBM PC, AT, XT, PCjr and most IBM compatibles.

PRINTQ™



- ★ DOUBLE PC OUTPUT.
- ★ ENHANCE PRINTER CONTROL.

Get the first real "mainframe" PC print spooler for just \$69.

PrintQ is a complete, independent print control subsystem that increases PC productivity so dramatically, it's like having an extra PC. Only better. PrintQ not only lets you run applications in the foreground while it controls your printer in the background, it actually gives you greater printer control than ever before. PrintQ gives you the power to:

- Re-start at any page after form jams
- Print up to 255 copies
- Get help via built-in text
- Minimize form changes
- View documents without printing

- Hold documents for later printing
- Simplify forms alignment
- Print in any order
- Control printer at any time via pop-up display
- Store as much data as your disk can hold
- Interface with existing programs
- Boot directly

Save time. Save money. Get maximum productivity from your PC and printer whether you're an advanced or beginner user. Call now or mail the coupon to order.

1-800-346-7638
In NJ, Call 201-361-1550

PRINTQ, Software Directions, Inc. 425 Route 10, Randolph, NJ 07869

YES. Rush me PrintQ for just \$69, which includes postage and handling. After using PrintQ, if I'm not convinced it increases my PC's productivity, I'll return it within 30 days for a full refund.

Name _____

Address _____

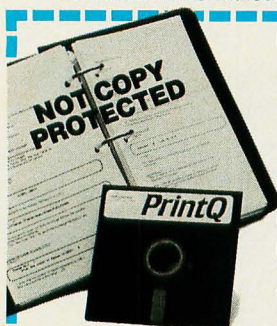
City _____ State _____ Zip _____

Check enclosed Visa Mastercard American Express

Acct. No. _____ Expiration Date _____

Signature _____

IBM is a registered trademark of International Business Machines Corporation. B10



NJ residents add 6% sales tax.