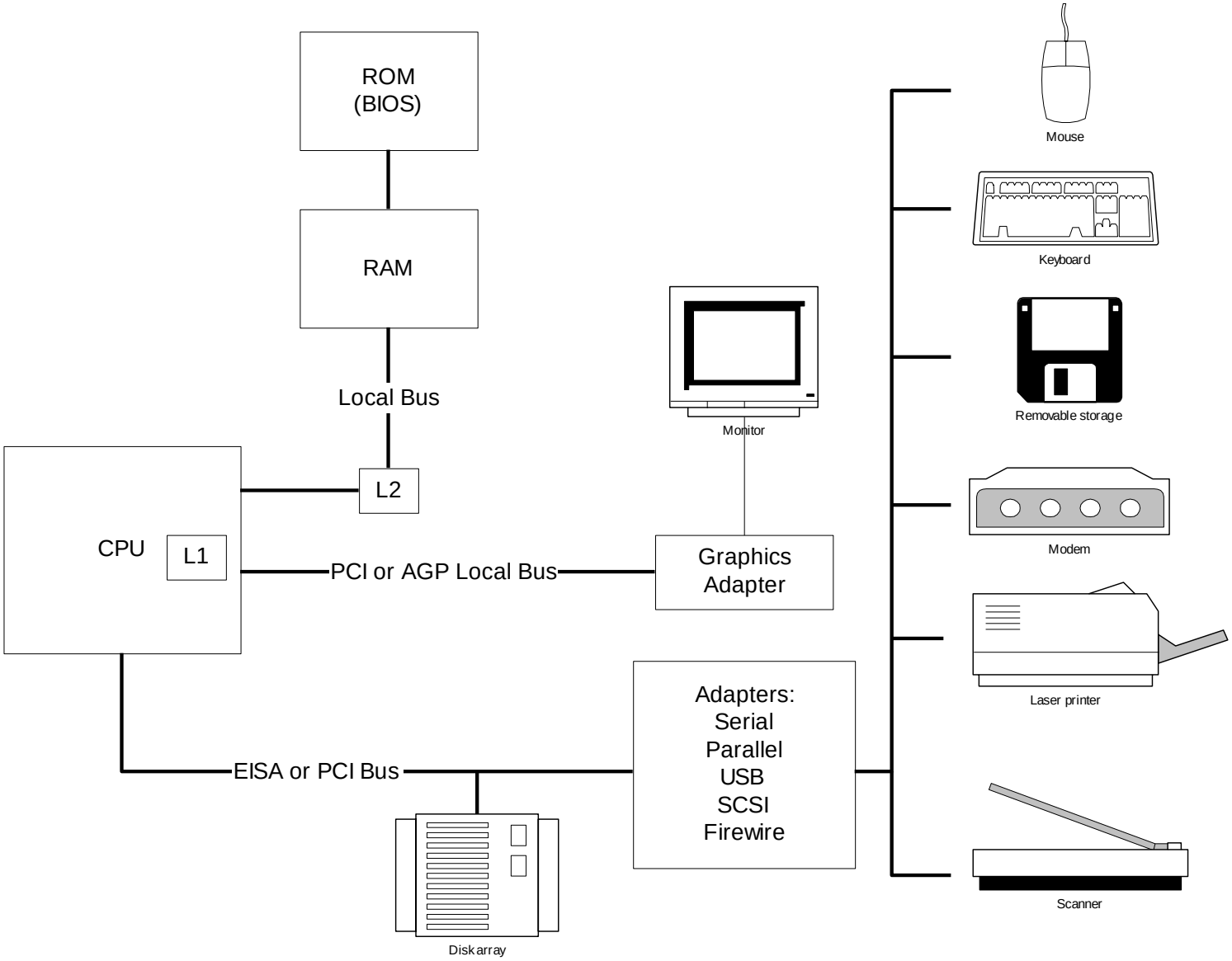
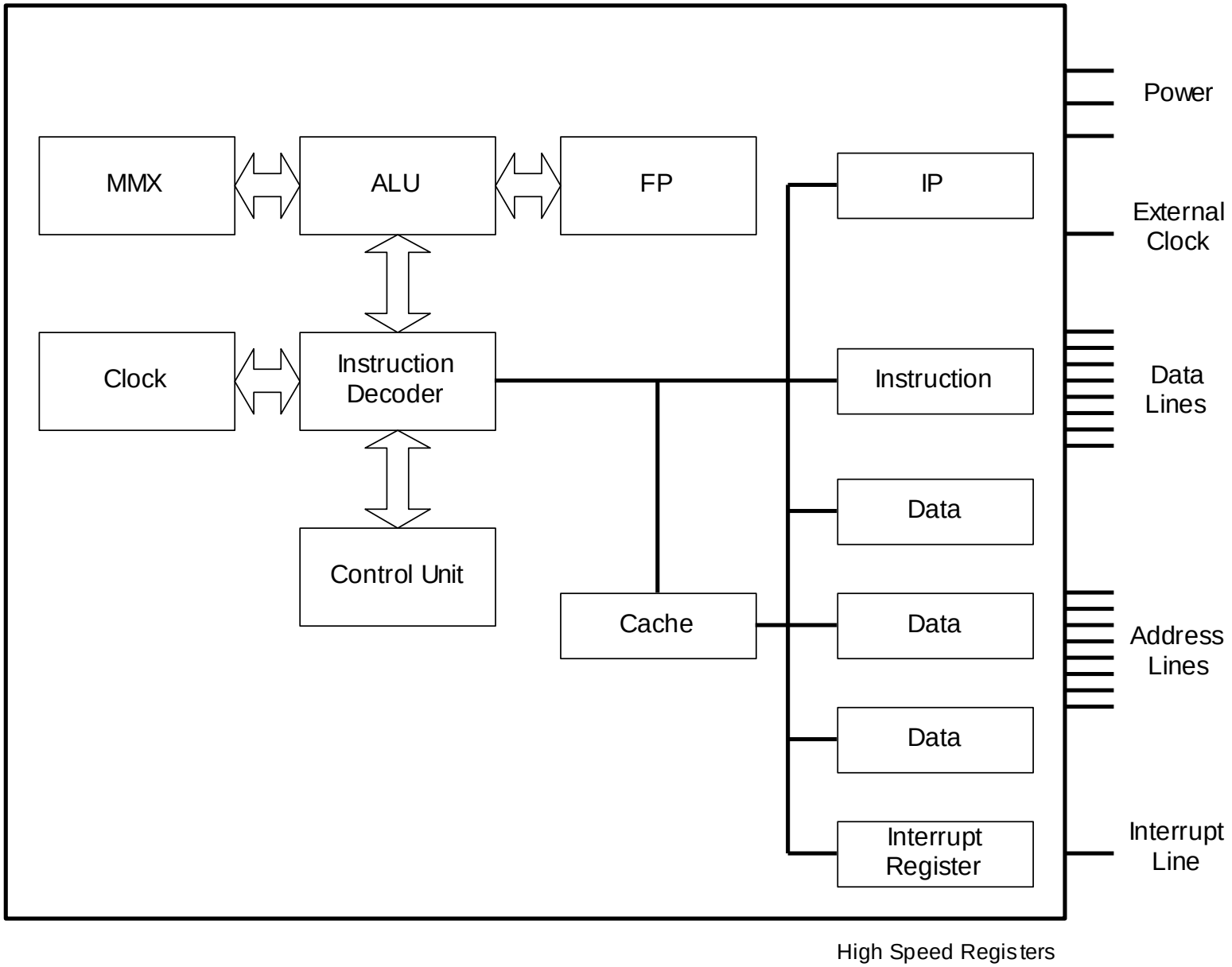
Computer Block Diagram



CPU

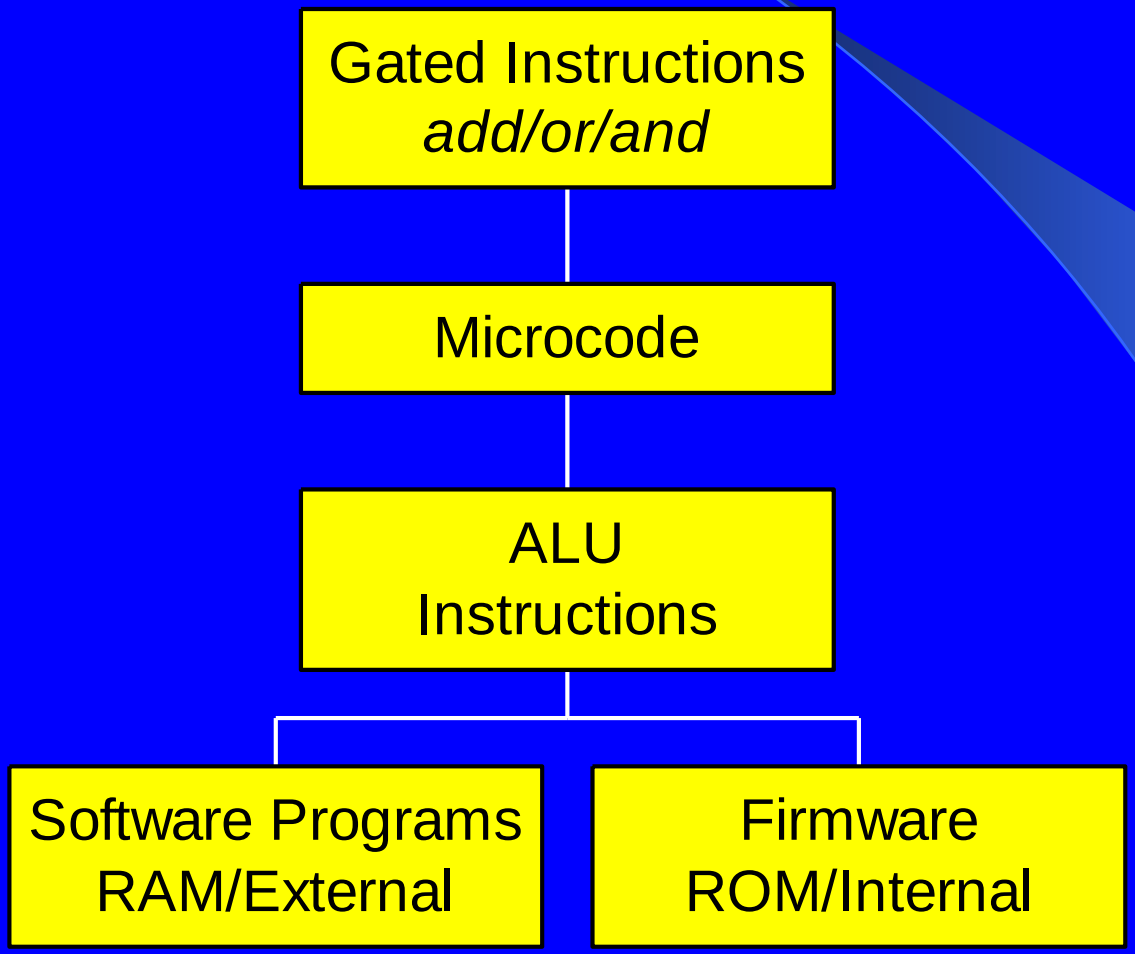

MMX
ALU
FP
Clock
Instruction Decoder
Control Unit
Cache
IP
Instruction
Data
Data
Data
Interrupt Register
Power
External Clock
Data Lines
Address Lines
Interrupt Line
High Speed Registers

Generation	Processor	Manu-facturer	Data Lines	Address Space			Cache Size	Copro-cessor	Clock Doubler
				Lines	Physical	Logical			
8088	8088	Various	8	20	1Mb	1Mb	-	-	-
	8086	Various	16	20	1Mb	1Mb	-	-	-
80286	80286	Various	16	24	16Mb	1Gb	-	-	-
80386	80386SX	Intel, AMD	16	24	16Mb	64Tb	-	-	-
	80386DX	Intel, AMD	32	32	4Gb	64Tb	-	-	-
	80386SLC	IBM	16	24	16Mb	64Tb	8Kb	-	-
80486	80486SX	Intel	32	32	4Gb	64Tb	8Kb	-	-
	80486DX	Intel	32	32	4Gb	64Tb	8Kb	Yes	-
	80486SX2	Intel	32	32	4Gb	64Tb	8Kb	-	Yes
	80486DX2	Intel	32	32	4Gb	64Tb	8Kb	Yes	Yes
	80486SLC2	IBM	16	24	16Mb	64Tb	16Kb	-	Yes
	80486SLC	Cyrix	16	24	16Mb	64Tb	1Kb	-	-
	80486DLC	Cyrix	32	32	4Gb	64Tb	1Kb	-	-

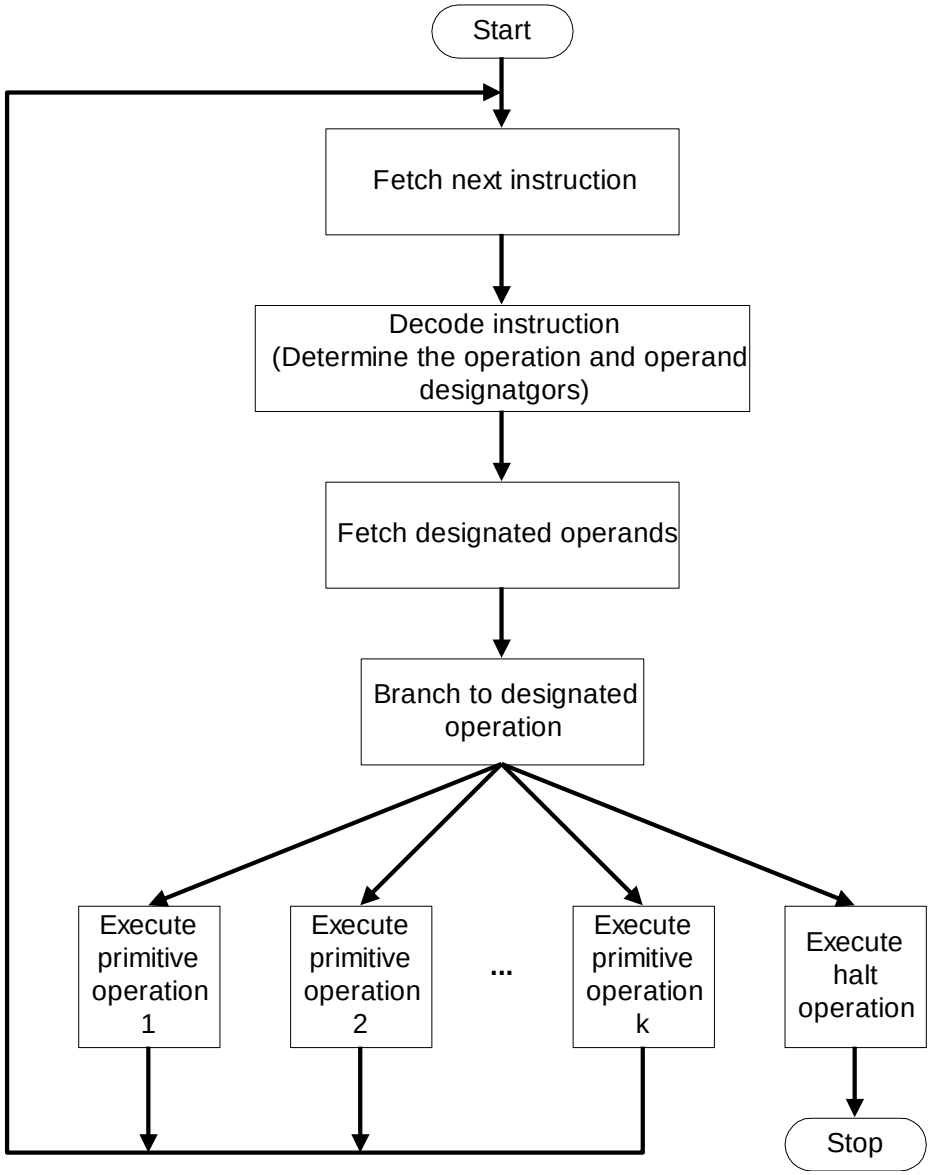
The range of PC processors currently available

Name	Date	Transistors	Microns	Clock speed	Data width	MIPS
8080	1974	6,000	6	2 MHz	8 bits	0.64
8088	1979	29,000	3	5 MHz	16 bits, 8-bit bus	0.33
80286	1982	134,000	1.5	6 MHz	16 bits	1
80386	1985	275,000	1.5	16 MHz	32 bits	5
80486	1989	1,200,000	1	25 MHz	32 bits	20
Pentium	1993	3,100,000	0.8	60 MHz	32 bits, 64-bit bus	100
Pentium II	1997	7,500,000	0.35	233 MHz	32 bits, 64-bit bus	~300
Pentium III	1999	9,500,000	0.25	450 MHz	32 bits, 64-bit bus	~510
Pentium 4	2000	42,000,000	0.18	1.5 GHz	32 bits, 64-bit bus	~1,700

Instruction Hierarchy



Basic Procedure for Program Interpretation & Execution



Role of an Operating System

- User interface
  - GUI
- Application interface
- Memory Management
- Time Management
  - Multitasking, virtual machine handler
- File System Manager
- Programming interface
  - API's, ActiveX
- Hardware Interface
  - Device Drivers, VxDs
- Multimedia Framework
  - Video for Windows, DirectX

Space and Time Management

- Intel Hardware Processing Modes
  - Real mode
  - Protected mode
  - Virtual 8086 mode
- Operating System Multiprocessing Modes:
  - Time sharing
  - Cooperative multitasking
  - Preemptive multitasking

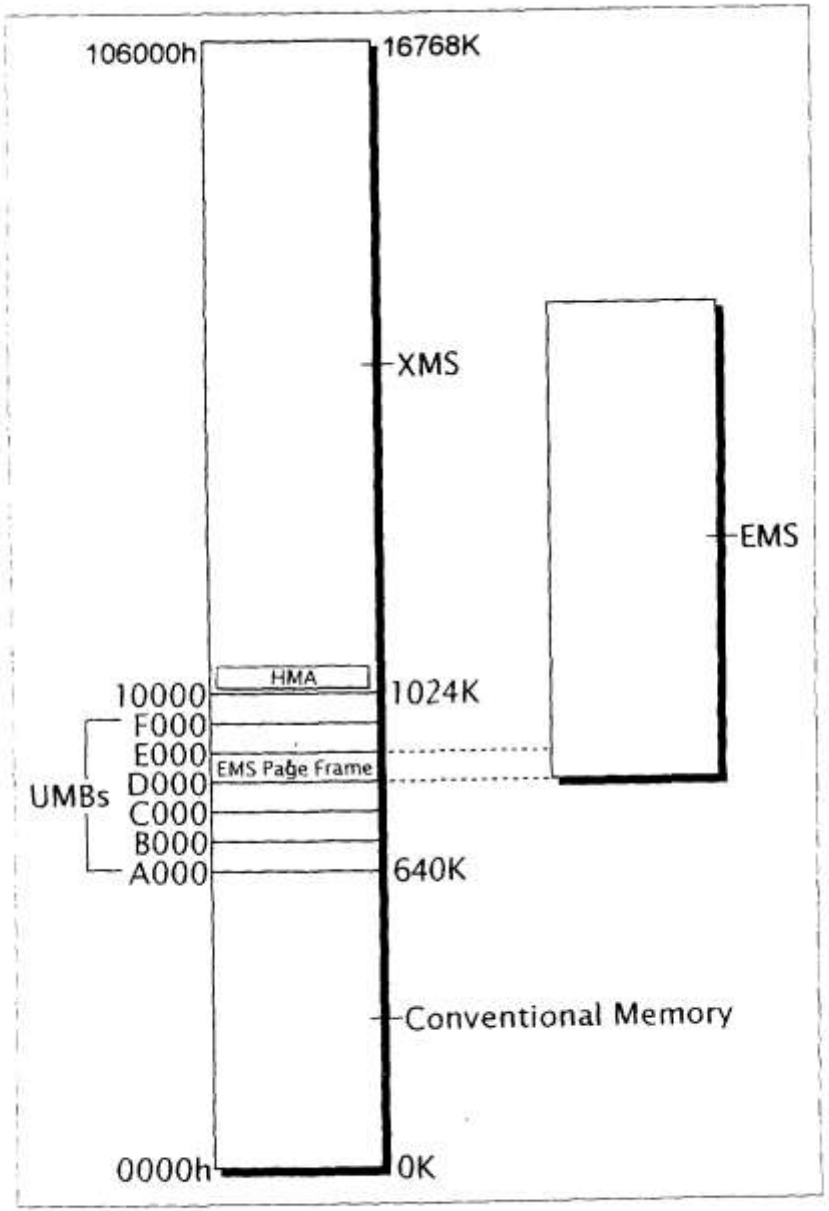


Figure 18-1: This diagram shows a PC with 16MB of RAM (640K of conventional memory plus 15744K of extended memory = 16384K plus 384K of UMBs, for a total of 16768K). EMS memory is taken from the extended memory by using EMM386.exe or other third-party memory managers, or is given to DOS programs running in Windows DOS sessions by Windows 95 itself.

Comparative OS Architectures: *The memory model of each operating*

WINDOWS 3.1

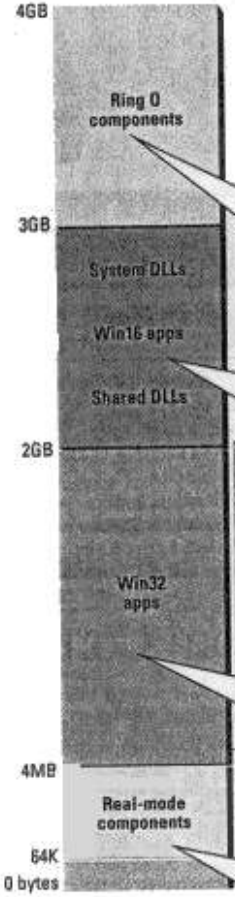
Windows 3.1 embodies trade-offs between performance and protection that hark back to the days of the 286. While it provides good performance for Win16 applica tions, DOS applications, real-mode device drivers, and virtual device drivers (VxDs it offers almost no protection against badly behaved applications.

The segmented memory space of a Win16 application is visible to and addressable by all other Win16 applications. Operating-system kernel DLLs and VxDs are also visible to all applications. This design, where a single pool of memory is shared among applications and the OS, provides fast performance as API calls to system DLLs do not entail ring transitions or context switches. The clear disadvantage is that any application can scribble on memory belonging to other applications or operating-system components, potentially bringing down the whole system.

Win16 applications are cooperatively multitasked. Under this system, the application, rather than the operating system, is responsible for giving up control to the next application. A crashed or misbehaving program can keep all others from receiving system resources.

WINDOWS 95

Windows 95 strikes a balance among performance, compatibility, and robustness. It offers fast execution of Win32, Win16, and DOS applications, and it can use real-mode device drivers. While it offers better crash protection than Windows 3.1, it remains vulnerable on several fronts.



Addresses between 2GB and 4GB are mapped into the address space of each Win32 application and are shared by all processes.

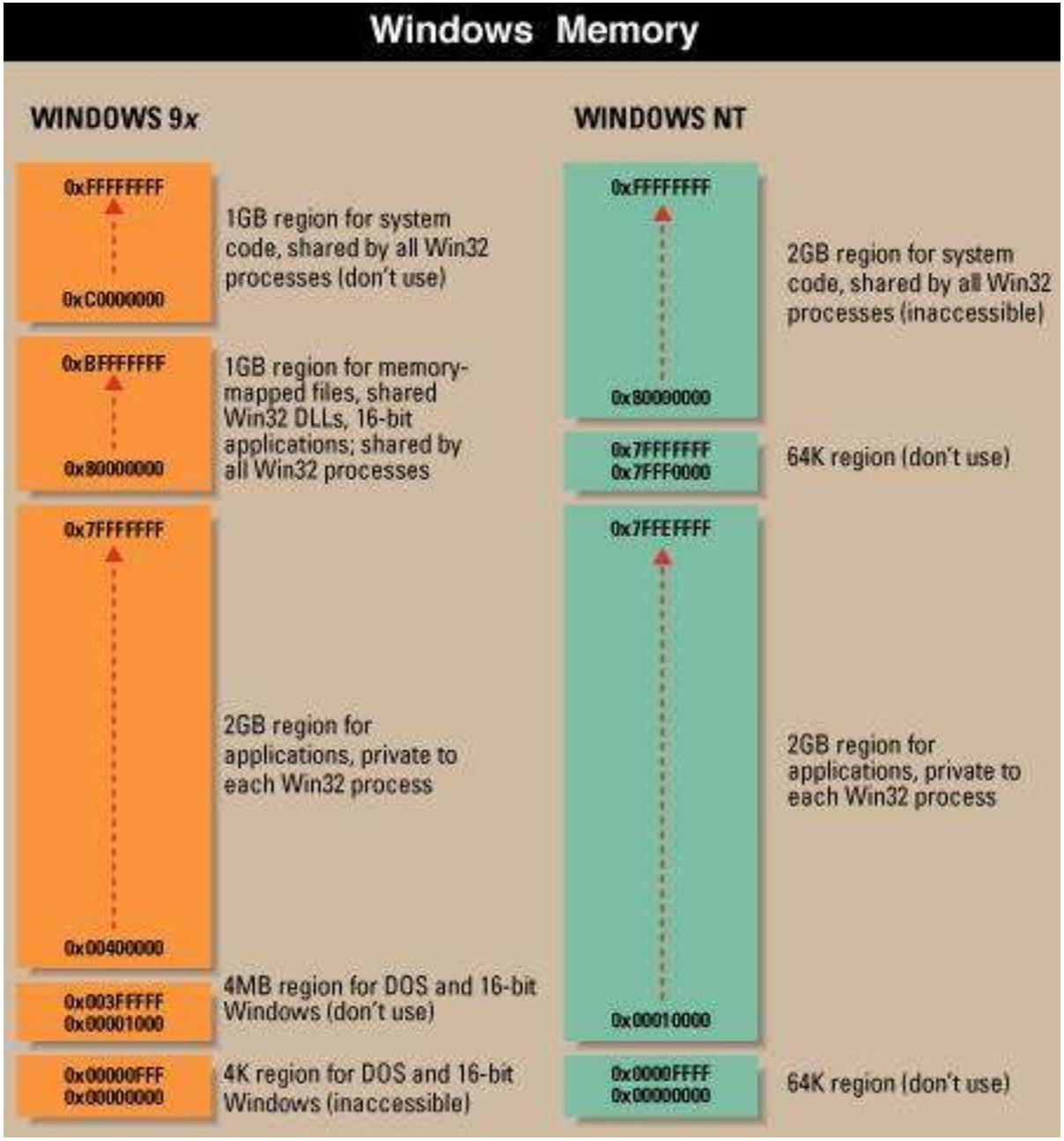
Addresses between 3GB and 4GB contain the Ring 0 components, including virtual device drivers (VxDs), the virtual machine management subsystem, and the file management subsystem. This memory area is visible to and writable by Win32 apps.

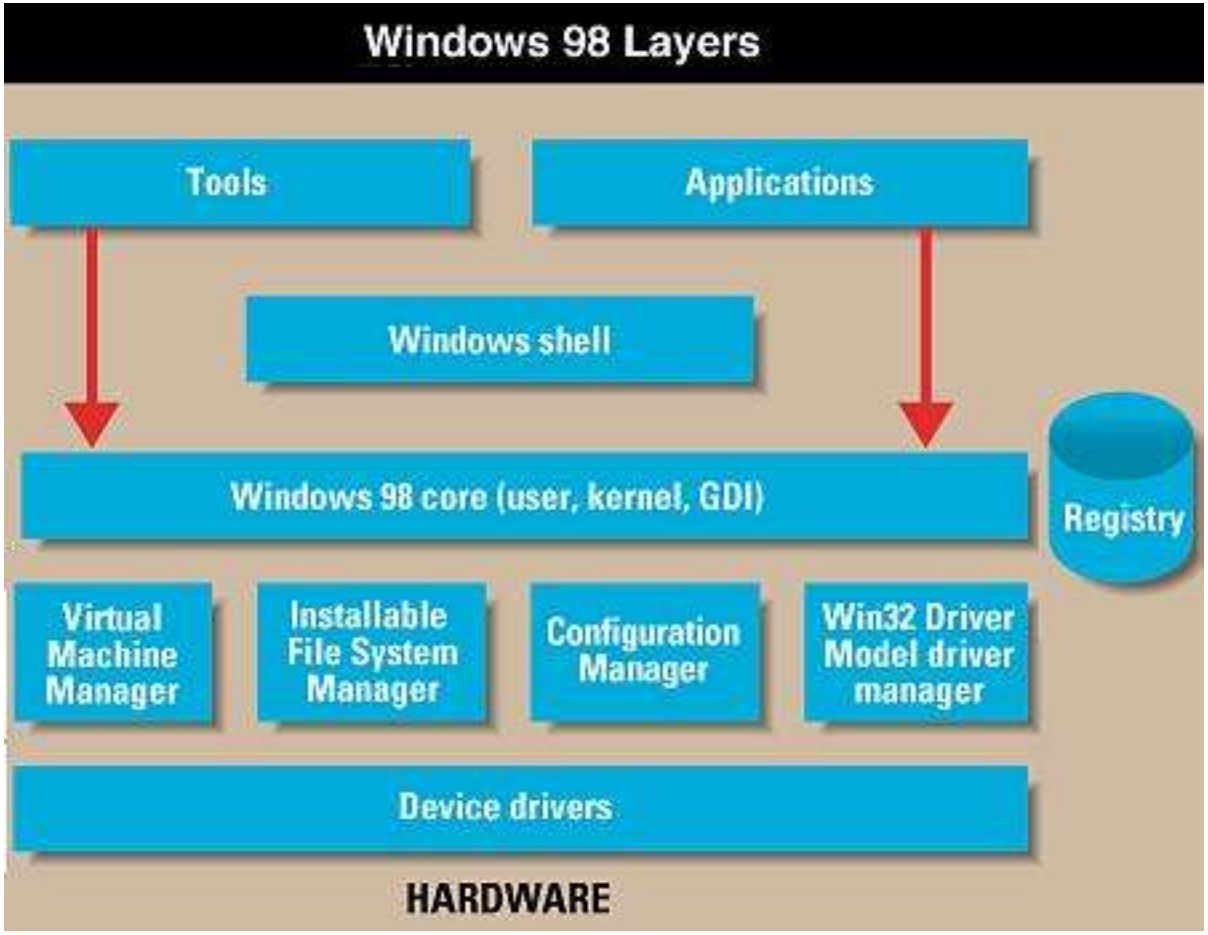
Addresses between 2GB and 3GB contain all Ring 3 Windows system DLLs, such as Kernel, GDI, and User, as well as any Win16 applications.

All 16-bit Windows applications run here in a shared address space, where they are cooperatively multitasked. This virtual 16-bit Windows machine is subject to the same vulnerabilities as Windows 3.1.

Between 4MB and 2GB, each Win32 app sees a private flat address space; it cannot see any other Win32 processes. Errors occurring here can usually be trapped without bringing down the operating system.

Addresses from 0 bytes to 4MB are shared by all processes. This is required for compatibility with real-mode device drivers, TSRs, and some 16-bit Windows applications, but makes it possible for any process to corrupt components residing in those addresses. While the first 64K is not addressable by Win32 applications, it can be addressed and potentially corrupted by 16-bit applications.


Windows Memory
WINDOWS 9x
0xFFFFFFFF
0xC0000000
1GB region for system code, shared by all Win32 processes (don't use)
0xBFFFFFFF
0x80000000
1GB region for memory-mapped files, shared Win32 DLLs, 16-bit applications; shared by all Win32 processes
0x7FFFFFFF
0x00400000
2GB region for applications, private to each Win32 process
0x003FFFFF
0x00001000
4MB region for DOS and 16-bit Windows (don't use)
0x00000FFF
0x00000000
4K region for DOS and 16-bit Windows (inaccessible)
WINDOWS NT
0xFFFFFFFF
0x80000000
2GB region for system code, shared by all Win32 processes (inaccessible)
0x7FFFFFFF
0x7FFF0000
64K region (don't use)
0x7FFEFFFF
0x00010000
2GB region for applications, private to each Win32 process
0x0000FFFF
0x00000000
64K region (don't use)


Windows 98 Layers
Tools
Applications
Windows shell
Windows 98 core (user, kernel, GDI)
Registry
Virtual Machine Manager
Installable File System Manager
Configuration Manager
Win32 Driver Model driver manager
Device drivers
HARDWARE