

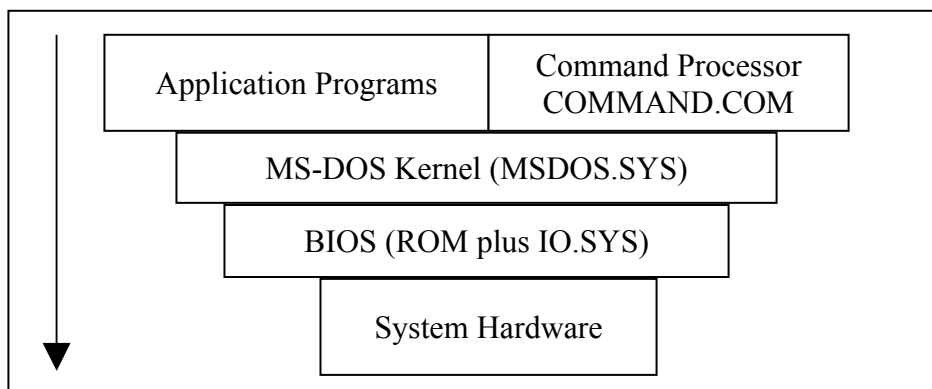
## DOS and BIOS interrupts

Many software programs written for 80x86 computers are designed to run under the MS-DOS operating system. Included as part of this operating system are the DOS functions and BIOS calls. These are subprograms, callable from applications software that can be used to access the hardware to the PC.

### The Structure of MS-DOS

The bios routines are the most primitive in a computer as they “talk” directly to the system hardware. Accordingly, the BIOS is hardware specific: that is, it must know the exact port address and control bit configurations for each I/O device in the computer. Typically, the BIOS is supplied by the computer manufacturer and resides in one or two ROM chips on the system board of the computer.

The BIOS interacts directly with the system hardware. The MS-DOS kernel accepts requests from the applications programs and passes these on to the BIOS and system hardware.



Levels of computing from Hardware to Application programs in MS-DOS

The resident **BIOS** which is stored in ROM and the IO.SYS file are loaded into RAM. The file IO.SYS is the extension for bios. These two are hardware dependent programs.

**The MS-DOS kernel:** In between the BIOS and the high-level applications software is the MS-DOS kernel. It is loaded into RAM when the system is booted via the file MSDOS.SYS (IBMDOS.SYS in PC-DOS). Unlike, the BIOS, the kernel provides hardware-independent functions.

For example, function 39 is used to create a disk subdirectory. To use this function the user ‘points’ register DX at the directory name (stored as a string of characters in memory), loads register AH with 39h (the function number) and then executes the software interrupt INT 21H.

## 80x86 Interrupts

An interrupt is an event that causes the processor to suspend its present task and transfer control to a new program called the interrupt service routine (ISR). These are three sources of interrupts in an 80x86 computer: (1) processor interrupts, (2) software interrupts and (3) hardware interrupts.

Each interrupt must supply a type number which is used by the processor as a pointer into an interrupt vector table (IVT) to determine the address of the interrupt’s service routine.

**Processor interrupts**(Type0 – 12 ). These interrupts are generated by the processor itself, usually in response to an error condition.

**Software Interrupts.** (Type 3 and Type 4). These are special 80x86 instructions that trigger an interrupt response from the processor.

**Hardware Interrupts:** The 80x86 processors have two pins reserved for this purpose. NMI (non-maskable Interrupt) and INTR. The INTR pin allows external h/w to activate any of the 256 interrupt types. With “INT type number” and MOV AH, “function number” the corresponding interrupt is executed. The Programmable Interrupt Controller 8259 is used to connect the different peripherals with interrupt service request lines.

Interrupt Vector Number	Description	Interrupt Initiated by			
		CPU	BIOS	MS-DOS	PC

0	Divide Error	All			
1	Single-Step	All			
2	Non-Maskable Interrupt	All			
3	Breakpoint	All			
4	Interrupt on overflow	All			
5	Bound	>= 286			
	Print Screen				
6	Invalid op-code	>=286			
7	Coprocessor not available	286,386			
7	Device not available	>=486			
8	Double fault Timer	>=286			IRQ0
9	CoProcessor segment overrun	286,386			IRQ1
	Reserved	>=486			
	Keyboard				
A	Invalid task state segment	>=386			
	Video				IRQ2
B	Segment not present	>=386			
	Serial Port 2				IRQ3
C	State Fault	>=386			
	Serial Port1				IRQ4
D	General Protection Port				
	Second Parallel Port				IRQ5
E	Page Fault	>=386			
	Floppy Disk				IRQ6
F	Reserved	X			
	First Parallel Port				IRQ7
10	Coprocessor Error	>=286			
	Video Services		X		
11	Alignment Check	>=486			
	Equipment List service		X		
12	Machine check	>=Pentium	X		
	Machine Size Service				
13	Disk services		X		
14	Communications services		X		
15	System services		X		
16	Standard Keyboard services		X		
17	Printer services		X		
18	Activate ROM BASIC		X		
19	Activate bootstrap start-up routine		X		
1A	Time and data service		X		
1B-1F	PC hardware specific				X
20	Program terminate			X	
21	General MS-DOS Services			X	
22	Terminate address			X	
23	Control-C handler			X	
24	Critical error handler address			X	
25	Absolute disk read			X	
26	Absolute disk write			X	
27	Terminate and stay resident			X	
28	MS-DOS idle interrupt			X	

29-2E	MS-DOS internal use			X	
2F	Multiplex interrupt			X	
30-3F	MS-DOS internal use			X	
40-FF	PC hardware specific				IRQ8-15