

IA-32 architecture exceptions

exceptions						
#	ID	description	type #1		error code	source(s)
			EIP	B/C		
00h	#DE	divide error	fault	C #2	no	AAM/DIV/IDIV divide by zero, DIV/IDIV result too large
01h	#DB	debug	fault	B	DR6	DR0..3 code BP, DR7.GD=1
			trap			TSS.T=1, FLAGS.TF=1, INT1, DR0..3 data BP, DR0..3 I/O BP
02h	NMI	non-maskable interrupt	n/a	(B)	no	non-maskable interrupt
03h	#BP	breakpoint	trap	B	no	INT3
04h	#OF	overflow	trap	B	no	INTO if FLAGS.OF=1
05h	#BR	boundary range exceeded	fault	B	no	BOUND if range exceeded
06h	#UD	undefined opcode	fault	B	no	undefined opcode #6, illegal LOCK, RSM outside SMM, MOV from/to DR4..5 if CR4.DE=1, LLDT/SLDT/LTR/STR/LSL/LAR/VERR/VERW/ARPL outside PM MMX/SSE(x) if CR0.EM=1 (excludes SSE(x)-MEM) SSE(x) if CR4.OSFXSR=0 (excludes SSE(x)-MEM/MMX-SSE(x)) #XF if CR4.OSXMEX=0
07h	#NM	device not available	fault	B	no	WAIT if CR0.TS=1 and CR0.MP=1 FPU/FXSR if CR0.TS=1 or CR0.EM=1 #6 MMX/SSE(x) if CR0.TS=1 (excludes SSE(x)-MEM)
08h	#DF	double fault	abort	#3	0h	exceptions during exception handler invocation
09h	n/a	reserved	abort	n/a	no	was 80387 coprocessor segment overrun
0Ah	#TS	invalid TSS	fault	C	yes	implicit TSS accesses
0Bh	#NP	not present	fault	C	yes	segment register loads, explicit/implicit segment register accesses
0Ch	#SS	stack segment	fault	C	yes	SS loads, explicit/implicit SS accesses
0Dh	#GP	general protection	fault	C	yes	various
0Eh	#PF	page fault	fault	#4	CR2	not present page (PDPTE/PDE/PTE.P=0), supervisor page (CPL=3 read/write, PDE/PTE.US=0), read-only page (CPL=3 write, PDE/PTE.RW=0), write-protected page (CPL<3 write, CR0.WP=1, PDE/PTE.RW=0), reserved bit (CR4.PSE=1 or CR4.PAE=1, PDE/PTE.RSVD=1)
0Fh	n/a	reserved	n/a	n/a	n/a	All spurious IRQs generate IRQ7, and can not be blocked by the PIC1's mask. The default base vector for PIC1 is set to 0Fh by the BIOS, so that spurious IRQs will invoke vector 0Fh. Hence it is not used for processor exceptions.
10h	#MF	math fault	fault	B	no	#IS for invalid stack or #IA for invalid arithmetic operation, #D for denormalized operand, #Z for divide by zero, #O for numeric overflow, #U for numeric underflow, #P for inexact result (precision)
11h	#AC	alignment checking	fault	B #5	0h	CR0.AM=1, EFLAGS.AC=1, CPL=3, misaligned accesses P4: MISC_EN.SLD=1, lock accross cache line or page boundary
12h	#MC	machine check	abort	B	MSRs	internal error, bus error, or bus error detected by external agent
13h	#XF	extended math fault	fault	B	no	#I for invalid arithmetic operation, #D for denormalized operand, #Z for divide by zero, #O for numeric overflow, #U for numeric underflow, #P for inexact result (precision)
14h	n/a	reserved	n/a	n/a	n/a	reserved
15h	n/a	reserved	n/a	n/a	n/a	reserved
16h	n/a	reserved	n/a	n/a	n/a	reserved
17h	n/a	reserved	n/a	n/a	n/a	reserved
18h	n/a	reserved	n/a	n/a	n/a	reserved
19h	n/a	reserved	n/a	n/a	n/a	reserved

1Ah	n/a	reserved	n/a	n/a	n/a	reserved
1Bh	n/a	reserved	n/a	n/a	n/a	reserved
1Ch	n/a	reserved	n/a	n/a	n/a	reserved
1Dh	n/a	reserved	n/a	n/a	n/a	reserved
1Eh	n/a	reserved	n/a	n/a	n/a	reserved
1Fh	n/a	reserved	n/a	n/a	n/a	reserved
notes	description					
#1	fault: before instruction, trap: after instruction, abort: imprecise, B=benign: can't cause #DF, C=contributory: can cause #DF					
#2	The correct behavior would be for #DE to be a benign exception, because #DE can't be caused by any handler invocation.					
#3	Any further contributory exceptions during the #DF handler invocation cause the processor to enter the shutdown state.					
#4	Any #PF during the invocation of the #PF handler causes the processor to generate a #DF.					
#5	The correct behavior would be for #AC to be a contributory exception, to avoid the possibility of endless #AC loops in CPL=3.					
#6	The correct behavior for undefined opcodes in the FPU range (D8h,00h...DFh,FFh) while CR0.EM=1 or CR0.TS=1 would be to cause #UD (like the Intel P4-core does) instead of causing #NM (like older processors do).					

