Control register

A **control register** is a processor register which changes or controls the general behavior of a CPU or other digital device. Common tasks performed by control registers include interrupt control, switching the addressing mode, paging control, and coprocessor control.

Control registers in x86 series

CR0

The CR0 register is 32 bits long on the 386 and higher processors. On x86-64 processors in long mode, it (and the other control registers) is 64 bits long. CR0 has various control flags that modify the basic operation of the processor.

Bit	Name	Full Name	Description
0	PE	Protected Mode Enable	If 1, system is in protected mode, else system is in real mode
1	MP	Monitor co-processor	Controls interaction of WAIT/FWAIT instructions with TS flag in CR0
2	EM	Emulation	If set, no x87 floating point unit present, if clear, x87 FPU present
3	TS	Task switched	Allows saving x87 task context upon a task switch only after x87 instruction used
4	ET	Extension type	On the 386, it allowed to specify whether the external math coprocessor was an 80287 or 80387
5	NE	Numeric error	Enable internal x87 floating point error reporting when set, else enables PC style x87 error detection
16	WP	Write protect	When set, the CPU can't write to read-only pages when privilege level is 0
18	AM	Alignment mask	Alignment check enabled if AM set, AC flag (in EFLAGS register) set, and privilege level is 3
29	NW	Not-write through	Globally enables/disable write-through caching
30	CD	Cache disable	Globally enables/disable the memory cache
31	PG	Paging	If 1, enable paging and use the CR3 register, else disable paging

CR1

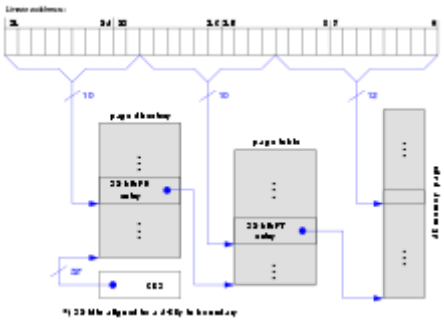
Reserved, the cpu will throw a #ud exception when trying to access them.

CR2

Contains a value called Page Fault Linear Address (PFLA). When a page fault occurs, the address the program attempted to access is stored in the CR2 register.

CR3

Used when virtual addressing is enabled, hence when the PG bit is set in CR0. CR3 enables the processor to translate linear addresses into physical addresses by locating the page directory and page tables for the current task. Typically, the upper 20 bits of CR3 become the *page directory base register* (PDBR), which stores the physical address of the first page directory entry.



Typical use of CR3 in address translation with 4 KiB pages.

CR4

Used in protected mode to control operations such as virtual-8086 support, enabling I/O breakpoints, page size extension and machine check exceptions.

Bit	Name	Full Name	Description
0	VME	Virtual 8086 Mode Extensions	If set, enables support for the virtual interrupt flag (VIF) in virtual-8086 mode.
1	PVI	Protected-mode Virtual Interrupts	If set, enables support for the virtual interrupt flag (VIF) in protected mode.
2	TSD	Time Stamp Disable	If set, RDTSC instruction can only be executed when in ring 0, otherwise RDTSC can be used at any privilege level.
3	DE	Debugging Extensions	If set, enables debug register based breaks on I/O space access
4	PSE	Page Size Extension	If unset, page size is 4 KiB, else page size is increased to 4 MiB (if PAE is enabled or the processor is in Long Mode this bit is ignored[1]).
5	PAE	Physical Address Extension	If set, changes page table layout to translate 32-bit virtual addresses into extended 36-bit physical addresses.
6	MCE	Machine Check Exception	If set, enables machine check interrupts to occur.
7	PGE	Page Global Enabled	If set, address translations (PDE or PTE records) may be shared between address spaces.
8	PCE	Performance-Monitoring Counter enable	If set, RDPMC can be executed at any privilege level, else RDPMC can only be used in ring 0.
9	OSFXSR	Operating system support for FXSAVE and FXRSTOR instructions	If set, enables SSE instructions and fast FPU save & restore
10	OSXMMEXCPT	Operating System Support for Unmasked SIMD Floating-Point Exceptions	If set, enables unmasked SSE exceptions.
13	VMXE	Virtual Machine Extensions Enable	see Intel VT-x
14	SMXE	Safer Mode Extensions Enable	see Trusted Execution Technology (TXT)
16	FSGSBASE	Enables the instructions RDFSBASE, RDGSBASE, WRFSBASE, and WRGSBASE.	
17	PCIDE	PCID Enable	If set, enables process-context identifiers (PCIDs).
18	OSXSAVE	XSAVE and Processor Extended States Enable	
20	SMEP[2]	Supervisor Mode Execution Protection Enable	If set, execution of code in a higher ring generates a fault
21	SMAP	Supervisor Mode Access Prevention Enable	If set, access of data in a higher ring generates a fault[3]
22	PKE	Protection Key Enable	See Intel® 64 and IA-32 Architectures Software Developer's Manual

CR5-7

Reserved, same case as CR1.

Additional Control registers in x86-64 series

EFER

Extended Feature Enable Register (EFER) is a model-specific register added in the AMD K6 processor, to allow enabling the SYSCALL/SYSRET instruction, and later for entering and exiting long mode. This register becomes architectural in AMD64 and has been adopted by Intel. Its MSR number is 0xC0000080.

Bit	Purpose
0	SCE (System Call Extensions)
1–7	Reserved
8	LME (Long Mode Enable)
9	Reserved
10	LMA (Long Mode Active)
11	NXE (No-Execute Enable)
12	SVME (Secure Virtual Machine Enable)
13	LMSLE (Long Mode Segment Limit Enable)
14	FFXSR (Fast FXSAVE/FXRSTOR)
15	TCE (Translation Cache Extension)
16–63	Reserved

CR8

CR8 is a new register accessible in 64-bit mode using the REX prefix. CR8 is used to prioritize external interrupts and is referred to as the task-priority register (TPR).[1]

The AMD64 architecture allows software to define up to 15 external interrupt-priority classes. Priority classes are numbered from 1 to 15, with priority-class 1 being the lowest and priority-class 15 the highest. CR8 uses the four low-order bits for specifying a task priority and the remaining 60 bits are reserved and must be written with zeros.

System software can use the TPR register to temporarily block low-priority interrupts from interrupting a high-priority task. This is accomplished by loading TPR with a value corresponding to the highest-priority interrupt that is to be blocked. For example, loading TPR with a value of 9 (1001b) blocks all interrupts with a priority class of 9 or less, while allowing all interrupts with a priority class of 10 or more to be recognized. Loading TPR with 0 enables all external interrupts. Loading TPR with 15 (1111b) disables all external interrupts.

The TPR is cleared to 0 on reset.

See also

- General purpose registers
- Test register
- Model-specific registers
- Debug register
- Flag byte
- Status register

References

1. "AMD64 Architecture Programmer's Manual Volume 2: System Programming" (http://developer.amd.com/wordpress/media/2012/10/24593_APM_v2.pdf) (PDF). AMD. September 2012. p. 127 & 130. Retrieved 2017-08-04.