bits	63-32	Array Pointer (edx)
	31-0	Array Data (eax)

Register 83h Hardware Configuration Register — Am5$_K$86

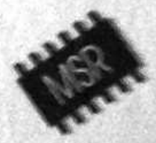
This register is used to control cache, branch tracing, debug, and clock control functions.

bits	63-8	Reserved (set to 0 for WRMSR)
	7 = 0	Enable Data Cache
	1	Disable Data Cache
	6 = 0	Enable Instruction Cache
	1	Disable Instruction Cache
	5 = 0	Enable Branch Prediction
	1	Disable Branch Prediction
	3-1 = x	Debug Control (nonspecified bit combinations reserved)

bit 3	bit 2	bit 1	
0	0	0	= Disable debug control
0	0	1	= Enable branch-tracing messages
1	0	0	= Activate probe mode on debug trap

	0 = 0	Enables stopping of CPU clocks in HALT and Stop Grant states
	1	Disables stopping of CPU clocks in HALT and Stop Grant states

Register 1000h Processor Operation Register — IBM 386/486SLC



This register controls a variety of custom features only available on these IBM chips. Bits 15 to 17 are available on the 486SLC but are reserved on the 386SLC. The functions for all the bits specified in EDX:EAX are:

bits	63-19	Reserved, and presumed unused
	18 = 1	Low power PLA mode—Additional dynamic parts of the CPU are powered down during the low-power halt state. (Unavailable or undocumented in 386SLC)
	17 = 1	Force all reads from external memory, even if cache is enabled. It is used for factory CPU testing. (Unavailable or undocumented in 386SLC)
	16 = 1	Switch the internal cache from odd to even parity. This forces the generation of an internal cache parity error. It is intended for factory CPU testing. (Unavailable or undocumented in 386SLC)

15 = 1 Enable Cacheability of floating point operand reads. If an external Intel FPU is used, this bit must be zero. If a Cyrix FPU is used, setting this bit on boosts opcode transfer performance.

14 = 0 Enable the ERROR input pin to Intel Compatible ERROR. The ERROR line is unused on all PC compatible designs, and was intended by Intel to signal errors from a math FPU.

1 Switch ERROR pin to a Hidden Memory Address Strobe output line. This is used to access hidden memory during Suspend or Emulation operations if supported in the motherboard. See the section on Suspend and Resume for more about hidden memory. This bit can only be set the first time WRMSR 1000h is issued after a hardware reset. Further attempts at changing the value are ignored to avoid an ERROR input clashing with the Strobe output option.

13 = 1 Enable Low Power Halt mode. A HALT instruction will cause the CPU to stop its internal clock to save power.

12 = 1 Wait for Ready after Output. After an output instruction is issued, the processor will wait until the CPU READY line is active before executing the next instruction. This allows working with devices that may be powered off and require extra time to come back online.

11 = 1 Cache Reload Status Bit—When an internal cache reload occurs, this bit is set by the CPU.

10 = 0 Internal Cache enable is determined by external hardware line input.

1 Internal Cache enable is determined within the CPU using the limits set from the Model specific Register 1001h. The pin that would be used as an input (when this flag is 0) becomes an output indicating whether a memory cycle is a cacheable cycle. This bit can only be set the first time WRMSR 1000h is issued after a hardware reset. Further attempts at changing the value are ignored.

9 = 1 Disable Cache Lock Mode—Allows the CPU to recognize a locked Read-Modify-Write cycle, but does not cache the cycle.

8 = x Reserved for unknown function or unused

7 = 1 Internal Cache Enabled—This is a similar function to the Cache Enable bit in the EFLAGS register of Intel's CPUs.

6 = 1 Disable caching for the region of memory E0000 to E0FFFh. This 4K area is used for Double Byte Character Support (DBCS) on a Japanese system, and should not be cached when DBCS is used.

5 = 1 Enable Power Interrupt PWI—Allows the PWI pin to control Suspend mode. See section on Suspend and Resume Modes. It also controls the number of bytes saved and restored. Also see LOADALL for additional information.

4 = 1 Enable Flush Snooping. This is used for specific motherboard designs. It flushes the internal CPU cache when the processor is in HOLD and a CPU signal line is activated. It can be used when bit 3 is zero.

3 = 1 Enable Snoop Input—When the CPU is in HOLD by hardware the CPU still monitors data on the bus. If a write occurs by another CPU or device to a memory location that is also in the cache, the cache item is invalidated.

2 = 1 Enable A20 Mask—The AT+ system must include some logic, normally external to the CPU, to mask the CPU address line A20. This is done to simulate an 8088 address space. This IBM CPU contains logic inside the CPU to optionally perform the same function. When this bit is on, address line 20 is disabled unless paging is active (paging is set in CR0 bit 31). When A20 mask bit is off, control is either external or the entire address range is accessible.

1 = 1 Cache Parity Enable—The internal CPU cache parity checking is enabled. If a parity error occurs, the cache is flushed, disabled (bit 7 is set to 0), parity error flag set (bit 0 is set to 1), and parity disabled (bit 1 is set to 0). The NMI handler is called to deal with the parity error.

0 = 1 Parity Error occurred in internal cache memory—Write a zero to clear this flag. See bit 1 for more details. This flag can be set by a parity error even if parity is disabled, but no actions are taken when parity is disabled.

Register 1001h Cache Region Control Registers — IBM 386/486SLC

When caching is enabled, the specific regions of physical memory that are allowed to be cached are set with this register. Caching also specifies any ROM memory areas, so the cache can ignore attempted writes into ROM space. Normally these registers are set by the BIOS POST operation based on the total amount and types of memory in the system. The 64 bits from EDX:EAX have the following contents:

bits 63-40 Reserved and presumed unused

39-32 Extended Memory Cache limit—Specifies the number of contiguous 64K blocks starting at the 1 MB boundary that can be cached. For example, a value of 0Fh indicates 15 megabytes are cacheable.

31-16 Read Only Cache blocks—Each bit represents a 64K region of memory in the first Megabyte of memory space that has ROM memory. Bit 31 is set to indicate the last 64K at segment F000h is ROM. Bit 30 is set to indicate the 64K block at E000h is ROM, and so forth. A write to an area set as ROM will not be updated in cache memory.

15-0 First Megabyte Cacheable—Each bit represents a 64K region of memory in the first megabyte of memory space that can be cached. Bit 15 is set to indicate the last 64K at segment F000h can be cached. Bit 0 is set to indicate the first 64K block at 0 that can be cached.

Register 1002h Processor Operation Register 2 — IBM 486SLC2

This register controls different clock modes to allow the internal CPU clock to run twice as fast as the external clock. It also controls the process of changing the external clock frequency.

Because the internal clock can run at twice the frequency as the outside clock, it cannot be changed on-the-fly for a turbo mode feature. The CPU must be advised that a clock frequency change is requested. This request can be made either by setting a software bit or from an external pin on the chip. The CPU will respond on a hardware line when it is acceptable to change the incoming clock speed. Once the input clock frequency change has been made, the software request or hardware request should be removed, so the CPU can resume the normal clock state.

bits 63-30 Reserved and presumed unused

29 = 1 Enable External Dynamic Frequency Shift—This option allows the motherboard to control the CPU clock frequency shift. The system BIOS should be the only one to control this function because it is dependent on the motherboard design.

28 = 1 Dynamic Frequency Shift Ready—When a request is made for a frequency shift by hardware or setting bit 27 or when the CPU is ready for the shift, this flag is set.

27 = 1 Dynamic Frequency Request—Setting this bit to 1 requests the CPU to prepare for a clock input frequency change. When bit 28 is set, the input clock can be changed.

26 = x Clock mode

25 = x

24 = x

bit 26	bit 25	bit 24
0	0	0 = divide incoming clock by 2 (same as a 386SX)
0	1	1 = use incoming clock, no divide, for doubling internal CPU speed
1	0	0 = 3:1 clock mode (unconfirmed)

23-0 Reserved and presumed unused

Register 1004h Processor Control Register — IBM 486SBL3

This register controls options on the IBM 486BL3, Blue Lightning CPU (unconfirmed).

bits 63-24 Reserved and presumed unused

23 = 0 DD1 hardware

1 DD0 hardware (for OS/2 boot)

22 = 0 MOV CR0 decode for DD0, DD1A, DD1B and DD1D hardware

1 MOV CR0 decode for DD1C hardware

21 = x Unknown

20 = 0 Cache remains on

1 Cache disabled when not in use (low-power mode)

19 = x	Unknown
18 = 0	NOP instruction cycles (DD0 uses 2 cycles, DD1 uses 3 cycles)
1	NOP instruction cycles (DD0 uses 3 cycles, DD1 uses 2 cycles)
17-0 = x	Unknown

Inst	Description	Processors	A B C
XBTS	Extract Bit String	Intel 80386 A step	

```
0Fh, A6h, r/m   xbts    regW, regW/memW, ax, cl
```

The extract bit string instruction takes a string of bits from the first operand and places them in the second operand.

This instruction only exists in the very first 80386, referred to as the A step. It was eliminated in the next step to make room for additional microcode. It has not been duplicated by any competitor, since its presence signifies a very old 80386 chip that has a few quirks.

The same opcode was re-used in the first 80486 for the CMPXCHG instruction. It was discovered that some applications written prior to the 80486 were checking for the presence of the XBTS instruction, and were falsely thinking the new 80486 was an old 80386 A step part. So Intel moved the CMPXCHG instruction to a new opcode on the 80486 B step of the part! The end result is that most developers find it's way too much trouble to figure all this out, and just avoid using the CMPXCHG instruction. Also see the IBTS instruction earlier in this chapter.

The word forms of this instruction will appear as dword forms depending on the current mode (16- or 32-bit), and if a size override prefix is used.

Hidden Address Space

Some versions of the 386 and later CPUs provide features for a Suspend mode and assist with In-Circuit-Emulation. These features are accomplished in part by using hidden or "alternate" memory, not normally accessible from any program. This can only be used when both the CPU supports hidden memory and the hardware on the motherboard is designed to support hidden memory. Currently IBM-manufactured CPUs have this feature (386SLC and 486SLC), as does the Intel 80386SL, 80486SL and Pentium parts, and some AMD, Cyrix and TI parts. Keep in mind that many IBM computers use non-IBM CPUs that do not support these features, and IBM has sold motherboards to other vendors, which include IBM-made CPUs.

In most cases, three undocumented instructions are used for the suspend and emulation features. These include the In-Circuit-Emulator breakpoint (ICEBP), User Move Register (UMOV), and Load All Registers (LOADALL). In addition, the features may be controlled by the Read and Write Model Specific Register (RDMSR and WRMSR) instructions. Also note the RSM instruction. All of these instructions are described in detail in the previous section on undocumented instructions.