System Management Mode Explained

Despite Common Functions, Implementation Details Differ

By Mark Thorson

Recent implementations of Intel-compatible CPUs have a system management mode (SMM) for supporting advanced power-reduction strategies. A system management mode can be found in AMD's Am386DXLV and Am386SXLV, Chips and Technologies' Super386, and Cyrix's Cx486SLC (when the B0 revision becomes available), as well as Intel's own 386SL. An SMM can be used to provide these capabilities:

- Shutdown of the CPU and system logic with the option of restarting the system transparently to software.
- Transparent shutdown and restart of individual peripherals, such as disk drives.
- Transparent emulation of peripherals.
- Hooks for external power-management mechanisms, such as peripheral activity monitors (i.e., timers which signal when a peripheral has not been accessed for a programmed length of time).

Retrofitting these power-reduction capabilities onto the 80x86 architecture demands a high degree of software transparency because the PC/AT environment is hostile to new uses of existing architectural features. Unlike Apple's Macintosh, in which most applications make access to critical system resources through calls to the system ROM, PC applications usually make direct access to system resources. Many applications and TSR (terminate and stay resident) utilities are very ill-behaved—e.g., they remap the memory space, make private uses of interrupt vectors, etc.

An SMM avoids this problem because it is an extension beyond the base architecture. An SMM has these features:

- An address space outside of the normal memory space. This is used for both the instruction code and data (such as the CPU register save area) of the power-management software.
- A special interrupt mechanism, outside of the mechanism which uses the interrupt controllers and the interrupt descriptor table.
- New instructions for interfacing to the SMM. For example, each implementation has a special return-from-interrupt instruction to go with the new interrupt mechanism.
- New pins for supporting external hardware triggers to invoke the SMM and for distinguishing cycles to the SMM address space from normal memory cycles. The latter gives the designer the option of implementing the SMM address space in a physically separate memory, such as battery-backed SRAM.

Each vendor has implemented their SMM in a slightly different way—different addresses, different opcodes, etc.—but functionally, they are all roughly equivalent. The biggest difference is that Intel's 386SL is designed to work with the power-management logic on its companion 82360 integrated system logic chip, while the other chips are designed to work with third-party chip set solutions.

SMM Address Space

While in SMM, an alternative memory space is enabled for the code and data of the SMM interrupt handler. On AMD's Am386SXLV and Am386DXLV, the entire address space becomes the SMM space; i.e. all memory references use the bus signals which distinguish a normal memory cycle from an SMM memory cycle. On the Intel, C&T, and Cyrix chips, only a subset of the address space is swapped with the SMM space.

Intel's 386SL/82360 combination has only two options: a 32K space located at 38000 (hex) or a 64K space located at 30000. Cyrix's Cx486SLC allows any size between 4K and 32M in powers of two; it also allows a 4G space. It requires the space to be at an address aligned to the block size, but within this restriction the base can be anywhere in the physical address space. C&T's Super386 allows any base address and any block size (up to 1M), specified with byte granularity.

To accommodate the widest variety of designs, access must be provided between each address space and each mode. Before SMM can be used, some designs will load the power-management software to a RAM-based SMM space from the BIOS ROM. This will occur in normal mode (either real-address mode or protected mode), so it requires access to the SMM space from normal mode. While SMM is in effect, some designs may copy the DRAM contents onto disk, so that the system can be restarted transparently after a hard power-down. This requires the reverse—i.e., access to the normal address space while in SMM.

AMD and C&T don't provide any processor support for access to SMM space from normal mode. This feature isn't needed when the SMM code space is mapped to a ROM, and it can be provided in external logic when a separate SRAM is used. Both Intel and Cyrix provide a register bit to enable the SMM space while running in normal mode.

From inside SMM, Intel and Cyrix default to the normal memory space for any address outside of the SMM region. AMD allows access to normal space by extending the instruction set; the four flavors of the UMOV instructions can be used to read or write any byte, word, or doubleword in normal space while executing in SMM. C&T directs any memory reference to the GS segment to normal space.

After initializing the SMM space (if held in RAM), Intel and Cyrix have a register bit which can be toggled to enable calls through the SMM interrupt. The register bit is part of a configuration register accessed through the I/O space. In the AMD and C&T designs, external SMM interrupts are unmaskable, so the SMM code must be ROM-based or the SMM interrupt must be masked by external logic until the SMM code is loaded.

SMM Entry and Exit

On the Intel and Cyrix chips, the only trigger for entry into SMM is the interrupt request input, but the other vendors support both hardware and software trigger sources. Intel allows software to program a timer in the 82360 which triggers an SMM interrupt when it expires, while AMD and C&T have extended the instruction set with SMM call instructions.

The sources of the SMM interrupt are likely to include activity monitors and traps for I/O. An activity monitor is a hardware timer which gets reloaded on events which indicate activity, and expiration of the timer indicates the corresponding device has gone idle. A typical configuration would have one system activity monitor for invoking CPU clock speed reduction (or other system power-management strategies) and a few peripheral activity monitors.

Traps for I/O are used to intercept access to peripherals in a power-down state. External logic (or, in Intel's case, logic in the 82360 companion chip) decodes the I/O cycle and asserts the SMM interrupt. The interrupt handler can then power-up the peripheral and re-run the intercepted I/O instruction.

The SMM interrupt always has priority over other interrupts, including NMI. On entry into SMM, the processor is placed into real mode (i.e., 8086 mode) and interrupts are disabled. On the Intel and C&T chips, this is actually an enhanced real mode which allows access to 32-bit addressing. On the AMD and Cyrix chips, if an SMM handler needs 32-bit addressing it enters protected mode.

Intel, AMD, and Cyrix allow interrupts to be enabled and serviced in SMM, but most implementations probably won't attempt to do this because the SMM handler only needs to perform simple, quick functions that aren't likely to disrupt interrupt processing.

The entry point for the SMM handler is provided by a mechanism outside of the interrupt descriptor table used for handling other interrupts. Intel and AMD have entry points fixed in hardware; Intel uses 38000 (hex) while AMD uses the reset vector, either FFFFFFF0 (Am386DXLV) or FFFFF0 (Am386SXLV). C&T and Cyrix have programmable SMM entry points.

On SMM entry, some part of the processor state is stored to a save area in the SMM memory space. Intel stores the processor state to the region from 3FFA8 (hex) to 3FFFF. AMD stores to a region comprising 60000 to 600CA and 60100 to 60126. C&T and Cyrix both have a programmable base address for the register save area.

The number of registers saved on SMM entry varies widely between vendors. This is an important consideration, because it directly affects the speed of an SMM call or return. AMD saves most of the CPU register set, many internal temporary registers, and both the visible and invisible parts of the segment registers; it performs a total of 53 32-bit register saves and eight 16-bit saves. Intel saves most of the programmer-visible registers, for a total of 14 32-bit saves and eight 16-bit saves (only the visible parts of the segment registers are saved).

C&T and Cyrix save only a minimal subset of the register set, which allows rapid SMM entry and exit while leaving the designer the option of implementing a more complete register save. C&T saves four 32-bit registers (EBX, EDX, EIP, and EFLAGS) and one 64-bit register (CS, including both the visible and invisible parts). Cyrix saves six 32-bit registers (previous EIP, current EIP, EFLAGS, CR0, DR7, and either ESI or EDI depending on whether a REP OUTS or REP INS instruction was interrupted) and 11 bytes describing both the visible and invisible parts of the CS register.

The SMM memory implementation also affects the speed of SMM calls. If the designer chooses to put the SMM in a separate physical memory, Intel's design (using the 82360 system logic chip) requires the SMM to be located on the 8-bit X-bus, which runs at the 8 MHz speed of the AT bus. AMD's chips ignore requests for pipelined addressing and dynamic bus sizing during SMM cycles, so the SMM memory must be unpipelined and either 32 bits (DX) or 16 bits (SX) wide. C&T and Cyrix don't impose any restrictions on SMM cycles; they allow the same capabilities as normal memory cycles.

A special instruction is used to return from the SMM handler. It restores the CPU state from the processor save area, which has the effect of re-enabling interrupts. C&T has two of these SMM return instructions, SRET and SRESUME, in which the latter automatically disables interrupts for one cycle. This supports I/O trapping, in which SMM interrupts break *before* the I/O instruction (i.e., the processor state is restored, as with a page fault). The SRESUME instruction allows the emulated I/O device to be powered-up and the trapped instruction to be re-executed without trig-

Vendor	Opcode (hex)	Mnemonic	Description
Intel	0FAA	RSM	Return from SMM interrupt handler
AMD	F1	SMI	Call SMM interrupt handler
	0F07	RES3	Return from SMM interrupt handler
	0F10	UMOV	Move byte from register to normal space
	0F11	UMOV	Move word or dword (with operand size prefix) from register to normal space
	0F12	UMOV	Move byte from normal space to register
	0F13	UMOV	Move word or dword (with operand size prefix) from normal space to register
C&T	0F18	SCALL	Call SMM interrupt handler
	0F19	SRET	Return from SMM interrupt handler
	0F1A	SRESUME	Return from SMM with interrupts disabled for one instruction
	0F1B	SVECTOR	Exit from SMM and issue a shutdown cycle
	0F1E	EPIC	Load one of the six interrupt or I/O traps
	0F3C	RARF1	Read from bank 1 of the register file (includes visible and invisible CPU registers)
	0F3D	RARF2	Read from bank 2 of the register file
	0F3E	RARF3	Read from bank 3 of the register file
	0FF0	LTLB	Load TLB with page table entry
	0FF1	RCT	Read cache tag
	0FF2	WCT	Write cache tag
	0FF3	RCD	Read cache data
	0FF4	WCD	Write cache data
	0FF5	RTLBPA	Read TLB data (physical address)
	0FF6	RTLBLA	Read TLB tag (linear address)
	0FF7	LCFG	Load configuration register
	0FF8	SCFG	Store configuration register
	0FF9	RGPR	Read general-purpose register or any bank of register file
	0FFA	RARF0	Read from bank 0 of the register file
	0FFB	RARFE	Read from extra bank of the register file
	0FFD	WGPR	Write general-purpose register or any bank of register file
	0FFE	WARFE	Write extra bank of the register file
Cyrix	0F78	SVDC	Save segment register
	0F79	RSDC	Restore segment register
	0F7A	SVLDT	Save LDT register
	0F7B	RLDT	Restore LDT register
	0F7C	SVTS	Save task state register
	0F7D	RSTS	Restore task state register
	0FAA	RSM	Return from SMM interrupt handler

Table 1. Opcodes for instruction set extensions.

gering the I/O trap a second time.

On the AMD and Cyrix chips, SMM interrupts break *after* the current instruction. If the interrupt was an I/O trap, the SMM handler must restore the EIP and any other registers which were disturbed by the instruction before it can be re-executed.

Intel actually breaks after the instruction, but there are special addresses in the SMM space which the handler can reference to inform the processor that instruction re-execution is desired. After making one of these references, the SMM return instruction (called RSM on the Intel chip) will pass control back to the interrupted instruction in a manner functionally equivalent to breaking before the instruction.

The AMD and Cyrix chips provide an image of the previous EIP in the register save area. This pointer references the interrupted instruction, so it can be used to restore the EIP and examine the instruction. An SMM handler for the AMD chips must deal with an additional complication—they don't always break on the first iteration of string I/O instructions (i.e., instructions with the REP opcode prefix). A bit in the register save area indicates whether the break occurred on the first or second iteration.

New Instructions and Pins

Table 1 shows the instruction set extensions implemented on Intel's 386SL, AMD's Am386DXLV and Am386SXLV, C&T's Super386, and Cyrix's Cx486SLC. C&T and Cyrix have more instructions because a complete register save requires software intervention; the extra instructions are used to access registers which are difficult or impossible to access using the normal instruction set.

On the Intel and C&T chips, attempts to execute the new instructions in normal mode result in an illegal instruction trap, except for the SCALL instruction on the C&T chips which is intended to be executed from normal mode. On the AMD and Cyrix chips, the new instructions don't trap, and their use is usually not recommended. An exception to this rule is that the SMM return instruction is useful for loading the CPU registers in a manner similar to the LOADALL instruction. AMD documents this use, and Cyrix is considering making this a documented part of their instruction set architecture.

Table 2 shows the pins on Intel's 386DX and 386SX which have been taken over by the non-Intel 386-compatible CPUs to implement SMM functions.

Vendor	Pin (DX-PQFP)	Pin (SX-PQFP)	Name	Description
AMD	59	43	SMI*	System management interrupt
	37	31	SMIADS*	System management address strobe
	36	30	SMIRDY*	System management ready
	58	29	IIBEN	I/O instruction break enable
C&T		43	ANMI#	System management interrupt
		31	AADS#	System management address strobe
Cyrix		47	SMI#	System management interrupt
		20	SMADS#	System management address strobe

Table 2. 386 pins redefined by non-Intel vendors.

Continued on page 21

by limiting their bus usage. If a proposed 32-bit design needs more bandwidth than the specified amount, it must implement a 64-bit data path. Similarly, if a module requires more than the specified amount of time to satisfy a read request, then it is required to split the transaction. The majority of these specifications are simple maximum-allowed timing requirements that place a performance floor on the implementation.

The two profiles that are nearing completion are P896.5, Profile M (Military), and P896.6, Profile T (Telecom). Profile M will become both an IEEE and military standard, and it has been driven by the U.S. Navy's Space and Naval Warfare Systems Command. Profile M is intended to address the needs of most Navy mission-critical computing applications in ships, submarines, aircraft, large missile, torpedo, and shore facilities.

To address the large range of physical environments, Profile M has three classes which use a common logical layer. The convection-cooled commercial class uses the same mechanical packaging as Profiles A, B and F. The intermediate class uses a smaller conduction-cooled board in severe environments where the ultimate in packaging density is not required. The final class is the most demanding and is based on the Standard Electronic Module–Format E (SEM-E). Due to its expense, this class is usually limited to tactical aircraft.

The Profile M logical layer has all the tools of a system bus with support for locks, cache coherency, and message passing. In addition, this profile has provisions for the detection of all single-bit and single-point errors and for dual redundant buses. The IEEE P1394 SerialBus is also used as an additional communication path. Additional backplane signals are defined that are unique to this application, such as nuclear event detect (!).

Profile T has been driven by several of the U.S., European, and Japanese telecommunications giants. Their application requires the protocols of a system bus in an environment that features continuous operation and existing equipment packaging standards. Single-bit and single-point error detection, redundant modules and buses, and live insertion and withdrawal are all provided for in the specification.

Finally, two new profiles have recently become official IEEE projects. P896.7, Profile C (Cable) will specify mechanisms for interconnecting Futurebus+ backplanes. P896.8, Profile D (Desktop) will specify a high performance parallel bus for the desktop and mezzanine environments. New applications continue to come forward. Space-based systems look promising as the next profile project. Clearly, the application profile approach has allowed and will continue to allow Futurebus+ to grow with the times and technologies. ♦

SMM Explained

Continued from page 16

All of these pins were no-connects, except for the IIBEN pin on the Am386DXLV, which is a V_{CC} pin on Intel's 386DX. Note that although AMD offers their Am386DX in a PGA package, this packaging option is not available for the Am386DXLV.

C&T's Super386 is offered in four forms, the 38600DX and 38600SX which are pin-compatible with Intel's parts, and the 38605DX and 38605SX which have additional signals for cache control and SMM. The DX version uses a larger package than the original 386DX, so only the SX version (which uses the same package as an Intel 386SX) is shown in the table.

Unlike the other vendors, AMD has a separate ready line for cycles to the SMM space, and an input to enable a mode which allows I/O instruction trapping. The IIBEN pin is checked on every I/O cycle, and if it is asserted the processor will not pipeline the execution of instructions immediately following an I/O instruction. This guarantees that no instructions will be executed before the SMM interrupt is called, although it is possible that instruction prefetches will occur.

Conclusion

Each vendor has implemented its SMM in a slightly different way, but all provide the basic functions needed to implement advanced power-management strategies. Intel's solution has the least flexibility (fixed addresses, fixed SMM width, and few SMM memory configuration options), but that is not surprising in a design which integrates the processor with the chip set logic. The 82360 could be replaced with proprietary logic, in which case the SMM space could be any size and width.

AMD's solution is notable for its inflexible SMM memory width and large processor save/restore context. Cyrix offers a simple, flexible, and efficient solution, which is supplemented by a separate suspend/resume mechanism that can be used with external clock stop logic to reduce power consumption. C&T's solution is the most sophisticated, with its on-chip interrupt and I/O trapping logic, wealth of conditions which can invoke SMM, and instruction backup facility.

The SMM in the processor chip will require some amount of support in the system logic chip set. Sophisticated system and peripheral power management will require external activity monitors to assert SMI# when a peripheral or the system has been idle for a certain length of time. External I/O trapping logic will be required to decode I/O cycles and selectively activate SMI# on cycles to devices which have been powered-down in a restartable state. Intel already has these features in their 82360; C&T has I/O trapping.♦