AMD
K86™ Family
BIOS Design

Application Note

Publication # **21329** Rev: **G** Amendment/**0**
Issue Date: **December 1997**

This document contains information on a product under development at Advanced Micro Devices (AMD). The information is intended to help you evaluate this product. AMD reserves the right to change or discontinue work on this proposed product without notice.

Advanced Micro Devices, Inc. (“AMD”) reserves the right to make changes in its products without notice in order to improve design or performance characteristics.

The information in this publication is believed to be accurate at the time of publication, but AMD makes no representations or warranties with respect to the accuracy or completeness of the contents of this publication or the information contained herein, and reserves the right to make changes at any time, without notice. AMD disclaims responsibility for any consequences resulting from the use of the information included in this publication.

This publication neither states nor implies any representations or warranties of any kind, including but not limited to, any implied warranty of merchantability or fitness for a particular purpose. AMD products are not authorized for use as critical components in life support devices or systems without AMD’s written approval. AMD assumes no liability whatsoever for claims associated with the sale or use (including the use of engineering samples) of AMD products, except as provided in AMD’s Terms and Conditions of Sale for such products.

Trademarks

AMD, the AMD logo, and combinations thereof, K86, AMD-K5, AMD-3D, the AMD-K6 logo, and AMD-640 are trademarks, and RISC86 and AMD-K6 are registered trademarks of Advanced Micro Devices, Inc.

MMX is a trademark and Pentium is a registered trademark of the Intel Corporation.

Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.

Contents

List of Figures

List of Tables

Revision History

Date	Rev	Description
Sept 1997	F	Added support for AMD-K6® enhanced processor Model 7, 8, and 9. See Table 4 on page 7.
Sept 1997	F	Removed support for AMD-K5™-PR150. See Table 3 on page 6.
Sept 1997	F	Modified SYSCALL/SYSRET Target Address (STAR) Model-Specific Register (MSR) description. See page 21.
Sept 1997	F	Modified the description of the WCDE bit the in Write Handling Control Register (WHCR) MSR. See page 22.
Sept 1997	F	Modified processor RESET and INIT state tables (EDX register) to allow for future models. See Tables 1 and 2 on page 4.
Dec 1997	G	Added boot string information for the AMD-K6 3D processor to CPUID description on page 3.
Dec 1997	G	Added processor speeds, bus speeds, and boot string information for the AMD-K6 3D processor to Table 4 on page 7.
Dec 1997	G	Added AMD-3D™ to "New AMD-K6® Processor Instructions" on page 24.

Application Note

AMD K86™ Family BIOS Design

This document highlights the BIOS modifications required to fully support the K86™ family of processors, which includes the AMD-K5™ processor and the AMD-K6® processor. This document is a supplement to the *AMD K86 Family BIOS and Software Tools Developers Guide*, order# 21062.

There can be more than one way to implement the functionality detailed in this document, and the information provided is for demonstration purposes.

Audience

It is assumed that the reader possesses the proper knowledge of the K86 processors, the x86 architecture, and programming requirements to understand the information presented in this document.

BIOS Consideration Checklist

AMD-K5™ Processor

CPUID

- Use the CPUID instruction to properly identify the AMD-K5 processor.
- Determine the processor type, stepping and features using functions 0000_0001h and 8000_0001h of the CPUID instruction.
- Boot-up display: The processor name is retrieved using CPUID extended functions 8000_0002h, 8000_0003h, and 8000_0004h. See “CPUID Identification Algorithms” on page 5 for more information.

CPU Speed Detection

- Use speed detection algorithms that do not rely on repetitive instruction sequences.
- Use the Time Stamp Counter (TSC) to ‘clock’ a timed operation and compare the result to the Real Time Clock (RTC) to determine the operating frequency. See the example of frequency-determination assembler code available on the AMD website at http://www.amd.com.
- Display the P-Rating that is shown in Table 3, “Summary of AMD-K5™ Processor CPU IDs and BIOS Boot Strings,” on page 6.

Model-Specific Registers (MSRs)

- Access only MSRs implemented in the AMD-K5 processor.
- Program the write allocate registers—Hardware Configuration Register (HWCR), Write Allocate Top-of-Memory and Control Register (WATMCR), and Write Allocate Programmable Memory Range Register (WAPMRR). See “Write Allocate Registers” on page 17 and the *Implementation of Write Allocate in the K86™ Processors Application Note*, order# 21326 for more information.

Cache Testing

- Perform cache testing on the AMD-K5 processor using the Array Access Register MSR. See “Array Access Register (AAR)” on page 13 for more information.

SMM Issues

- The System Management Mode (SMM) functionality of the AMD-K5 processor is the same as the Pentium® processor.
- Implement the AMD-K5 processor SMM state-save area in the same manner as Pentium except for the IDT Base and possibly Pentium-reserved areas. See “System Management Mode (SMM)” on page 10 for more information.

AMD-K6® Processor

CPUID

- Use the CPUID instruction to properly identify the AMD-K6 processor.
- Determine the processor type, stepping and features using functions 0000_0001h and 8000_0001h of the CPUID instruction.
- Boot-up display: The processor name should be displayed as 'AMD-K6(tm)/XXX' for Models 6 and 7 and 'AMD-K6 3D/XXX' for Model 8. For more information, see "CPUID Identification Algorithms" on page 5.

CPU Speed Detection

- Use speed detection algorithms that do not rely on repetitive instruction sequences.
- Use the Time Stamp Counter (TSC) to 'clock' a timed operation and compare the result to the Real Time Clock (RTC) to determine the operating frequency. See the example of frequency-determination assembler code available on the AMD website at http://www.amd.com.
- Display the recommended BIOS boot string as shown in Table 4, "Summary of AMD-K6® Processor CPU IDs and BIOS Boot Strings," on page 7.

Model-Specific Registers (MSRs)

- Only access MSRs implemented in the AMD-K6 processor.
- Enable Write Allocation by programming the Write Handling Control Register (WHCR). See "Write Handling Control Register (WHCR)" on page 22 and the *Implementation of Write Allocate in the K86™ Processors Application Note*, order# 21326 for more information.

Cache Testing

- Use the AMD-K6 processor's BIST function to test internal memories. See "Built-In Self-Test (BIST)" on page 9 for more information. The AMD-K6 does not contain MSRs to allow for cache testing.

SMM Issues

- The System Management Mode (SMM) functionality of the AMD-K6 processor is the same as the Pentium processor.
- Implement the AMD-K6 processor SMM state-save area in the same manner as Pentium except for the IDT Base and possibly Pentium-reserved areas. See "System Management Mode (SMM)" on page 10 for more information.

Processor RESET and INIT States

After the AMD-K5 processor and AMD-K6 processor have completed their initialization due to the recognition of an asserted RESET or INIT signal, the state of all architecture registers and Model-Specific Registers (MSRs) are compatible with Pentium. The only differences are listed in Tables 1 and 2.

Table 1. State of the AMD-K5™ Processor After RESET

Register	RESET State	Notes
GDTR	base:0000_0000 limit:0000h	
IDTR	base:0000_0000 limit:0000h	
EDX	0000_05MSh	3
AAR	0000_0000_0000_0000h	1
HWCR	0000_0000_0000_0000h	1
WATMCR	0000_0000_0000_0000h	1, 2
WAPMRR	0000_0000_000F_000Ah	1, 2

Notes:

1. *These MSRs are described in the "AMD K86 Family BIOS and Software Tools Developers Guide", order# 21062.*
2. *The AMD-K5 processor only supports write allocate on Models 1, 2, and 3, with a Stepping of 4 or greater.*
3. *M = Model, 050S-Model 0, 051S-Model 1, 052S-Model 2, 053S-Model 3.*
   S = Stepping, 05M4-Stepping 4.

Table 2. State of the AMD-K6® Processor After RESET

Register	RESET State	Notes
EDX	0000_05MXh	1
EFER	0000_0000_0000_0000h	2
STAR	0000_0000_0000_0000h	2
WHCR	0000_0000_0000_0000h	2

Notes:

1. *M = Model, 056X-Model 6, 057X-Model 7, 058X-Model 8, 059X-Model 9*
2. *These MSRs are described in the "AMD K86 Family BIOS and Software Tools Developers Guide", order# 21062.*

State of the K86™ Processors After INIT

The assertion of INIT causes the processor to empty its pipelines, initialize most of its internal state, and branch to address FFFF_FFF0h—the same instruction execution starting point used after RESET. Unlike RESET, the processor preserves the contents of its caches, the floating-point state, the SMM base, MSRs, and the CD and NW bits of the CR0 register.

The edge-sensitive interrupts FLUSH# and SMI# are sampled and preserved during the INIT process and are handled accordingly after the initialization is complete. However, the processor resets any pending NMI interrupt upon sampling INIT asserted.

INIT can be used as an accelerator for 80286 code that requires a reset to exit from Protected mode back to Real mode.

CPUID Identification Algorithms

The CPUID instruction provides complete information about the processor (vendor, type, name, etc.) and its capabilities (features). After detecting the processor and its capabilities, software can be accurately tuned to the system for maximum performance and benefit to users. For example, game software can test the performance level available from a particular processor by detecting the type of processor. If the performance level is high enough, the software can enable additional capabilities or more advanced algorithms. Another example involves testing if the processor supports MMX™ instructions. If the software finds this functionality present when it checks the feature bits, it can utilize these more powerful extensions for better performance on new multimedia software.

For more detailed information, refer to the *AMD Processor Recognition Application Note*, order# 20734, located at http://www.amd.com

Table 3 on page 6 and Table 4 on page 7 outline the family codes and model codes for the AMD K86 processors. Table 3 shows the CPU speed, the ‘P-Rating’, and the recommended BIOS boot-string associated with each AMD-K5 processor.

Table 4 on page 7 shows the recommended BIOS boot-string for the AMD-K6 processor. This recommended boot-string is 'AMD-K6(tm)/XXX' for Models 6 and 7 and 'AMD-K6 3D/XXX' for Model 8. The value for XXX is determined by calculating the core frequency of the processor. Use the Time Stamp Counter (TSC) to 'clock' a timed operation and compare the result to the Real Time Clock (RTC) to determine the operating frequency.

Note: *Table 3 and Table 4 on page 7 contain information intended to prepare the infrastructure for potential future products. These products may or may not be announced, but BIOS software should be prepared to support these options.*

Table 3. Summary of AMD-K5™ Processor CPU IDs and BIOS Boot Strings

Instruction Family Code	Model Code	CPU Speed (MHz)	CPU Bus Speed (MHz)	Recommended BIOS Boot-String	CPUID Functions 8000_0002, 3, 4 Return Values
5 (AMD-K5™ Processor)	0	75	50	AMD-K5-PR75	undefined
		90	60	AMD-K5-PR90	undefined
		100	66	AMD-K5-PR100	undefined
	1	90	60	AMD-K5-PR120	AMD-K5(tm) Processor
		100	66	AMD-K5-PR133	AMD-K5(tm) Processor
	2	116.7	66	AMD-K5-PR166	AMD-K5(tm) Processor
	3	133	66	AMD-K5-PR200	AMD-K5(tm) Processor

Table 4. Summary of AMD-K6® Processor CPU IDs and BIOS Boot Strings

Instruction Family Code	Model Code	CPU Speed (MHz)	CPU Bus Speed (MHz)	Recommended BIOS Boot-String Display
5 (AMD-K6® Processor)	6	166	66	AMD-K6(tm)/166
		200	66	AMD-K6(tm)/200
		233	66	AMD-K6(tm)/233
		266	66	AMD-K6(tm)/266
	7	166	66	AMD-K6(tm)/166
		200	66	AMD-K6(tm)/200
		233	66	AMD-K6(tm)/233
		266	66	AMD-K6(tm)/266
		300	66	AMD-K6(tm)/300
	8	200	66	AMD-K6 3D/200
		233	66	AMD-K6 3D/233
		266	66	AMD-K6 3D/266
		300	66	AMD-K6 3D/300
		333	66	AMD-K6 3D/333
		366	66	AMD-K6 3D/366
		400	66	AMD-K6 3D/400
		250	100	AMD-K6 3D/250
		300	100	AMD-K6 3D/300
		350	100	AMD-K6 3D/350
		400	100	AMD-K6 3D/400
		450	100	AMD-K6 3D/450
	9	TBD*	TBD*	TBD*

Note:

* *Until the release of AMD-K6 processors with a Model Code 9, use the default settings from the previous generation.*

Figure 1 shows a flow chart for the CPUID instruction. Use this chart to implement a CPUID algorithm.

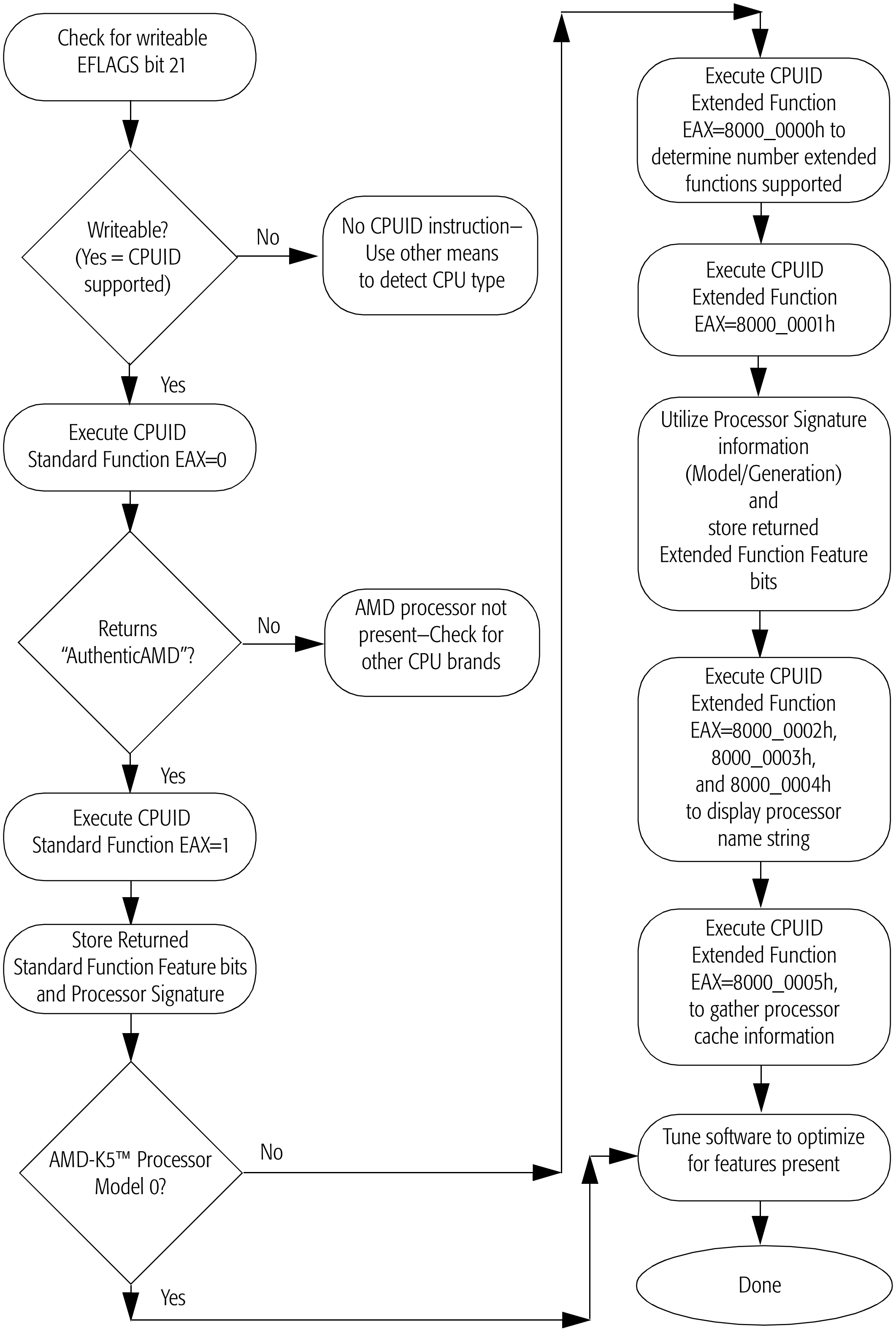

Check for writeable EFLAGS bit 21
Writeable? (Yes = CPUID supported)
No
No CPUID instruction–Use other means to detect CPU type
Yes
Execute CPUID Standard Function EAX=0
Returns "AuthenticAMD"?
No
AMD processor not present–Check for other CPU brands
Yes
Execute CPUID Standard Function EAX=1
Store Returned Standard Function Feature bits and Processor Signature
AMD-K5™ Processor Model 0?
No
Yes
Execute CPUID Extended Function EAX=8000_0000h to determine number extended functions supported
Execute CPUID Extended Function EAX=8000_0001h
Utilize Processor Signature information (Model/Generation) and store returned Extended Function Feature bits
Execute CPUID Extended Function EAX=8000_0002h, 8000_0003h, and 8000_0004h to display processor name string
Execute CPUID Extended Function EAX=8000_0005h, to gather processor cache information
Tune software to optimize for features present
Done

Figure 1. CPUID Instruction Flow Chart

Built-In Self-Test (BIST)

AMD-K5™ Processor

The BIST is invoked if INIT is asserted at the falling edge of RESET. Table 5 contains the results of BIST that are returned in the EAX register for the AMD-K5 processor.

Table 5. AMD-K5™ BIST Error Bit Definition in EAX Register

Bit Number	Bit Value	
	0	1
31–9	No Error	Always 0
8	No Error	Data path
7	No Error	Instruction-cache instructions
6	No Error	Instruction-cache linear tags
5	No Error	Data-cache linear tags
4	No Error	PLA
3	No Error	Microcode ROM
2	No Error	Data-cache data
1	No Error	Instruction cache physical tags
0	No Error	Data-cache physical tags

AMD-K6® Processor

For the AMD-K6 processor, BIST returns the result in the EAX general-purpose register after the completion of RESET. If EAX contains 0000_0000h, then BIST was successful. If EAX is non-zero, the BIST failed. BIST is run unconditionally after RESET.

System Management Mode (SMM)

This section documents the SMM differences between the AMD-K5 processor, AMD-K6 processor, and Pentium. For more information on SMM implementation in the K86 processors, see the *AMD K86 Family BIOS and Software Tools Developers Guide*, order# 21062.

State-Save Map Differences

The AMD-K5 processor and the AMD-K6 processor differ from Pentium in one way. The IDT Base location in the AMD-K5 processor and the AMD-K6 processor is located at offset FF90h. Pentium has the IDT Base located at offset FF94h.

I/O Trap Dword Differences

For both the AMD-K5 processor and AMD-K6 processor, the I/O trap dword is located at offset FFA4h. This state-save area, which is reserved in Pentium, contains information regarding an I/O instruction that may have been trapped by an SMI# assertion.

Table 6. AMD-K5™ Processor I/O Trap Dword Field at Offset FFA4h

Bits 31–16	Bit 15	Bit 14–2	Bit 1	Bit 0
I/O Port Address	I/O String Operation	Reserved	Valid I/O Instruction	Input or Output

Table 7. AMD-K6® Processor I/O Trap Dword Configuration at Offset FFA4h

Bits 31–16	Bits 15–4	Bit 3	Bit 2	Bit 1	Bit 0
I/O Port Address	Reserved	Rep String Operation	I/O String Operation	Valid I/O Instruction	Input or Output

Model-Specific Registers (MSRs)

The AMD-K5 processor and AMD-K6 processor support different MSRs. The two following sections describe the specific MSRs in each processor. For more details about the K86 MSRs, refer to the *AMD K86 Family BIOS and Software Tools Developers Guide*, order# 21062.

AMD-K5™ Processor

The AMD-K5 processor supports MSRs that can be accessed with the RDMSR and WRMSR instructions when CPL = 0. The following index values in the ECX register access specific MSRs:

- Machine-Check Address Register (MCAR)—ECX = 00h
- Machine-Check Type Register (MCTR)—ECX = 01h
- Time Stamp Counter (TSC)—ECX = 10h
- Array Access Register (AAR)—ECX = 82h
- Hardware Configuration Register (HWCR)—ECX = 83h
- Write Allocate Top-of-Memory and Control Register (WATMCR)—ECX = 85h
- Write Allocate Programmable Memory Range Register (WAPMRR)—ECX = 86h

Note: *The AMD-K5 processor supports write allocate only on Models 1, 2, and 3, with a Stepping of 4 or greater.*

Machine-Check Address Register (MCAR)

The processor latches the address of the current bus cycle in its 64-bit Machine-Check Address Register (MCAR) when a bus-cycle error occurs. These errors are indicated either by (a) system logic asserting BUSCHK, or (b) the processor asserting PCHK while system logic asserts PEN.

The MCAR can be read with the RDMSR instruction when the ECX register contains the value 00h. Figure 2 shows the format of the MCAR register.

If system software has set the MCE bit in CR4 before the bus-cycle error, the processor also generates a machine-check exception.

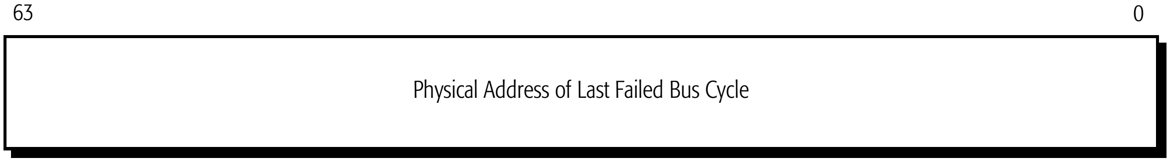


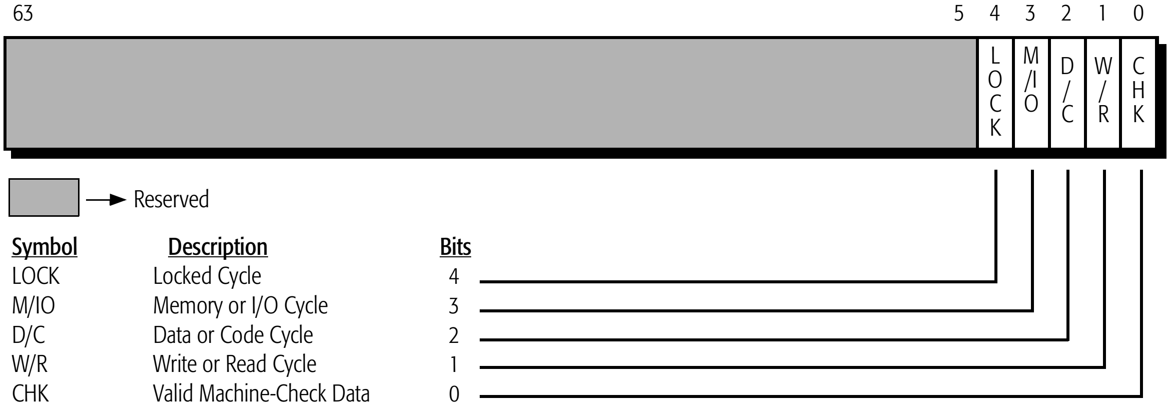
Figure 2. Machine-Check Address Register (MCAR)

Machine-Check Type Register (MCTR)

The processor latches the cycle definition and other information about the current bus cycle in its 64-bit Machine-Check Type Register (MCAR) at the same time that the Machine-Check Address Register (MCAR) latches the cycle address—when a bus-cycle error occurs. These errors are indicated either by (a) system logic asserting BUSCHK, or (b) the processor asserting PCHK while system logic asserts PEN.

The MCTR can be read with the RDMSR instruction when the ECX register contains the value 01h. Figure 3 and Table 8 show the formats of the MCTR register. The processor clears the CHK bit (bit 0) in MCTR when the register is read with the RDMSR instruction.

If system software has set the MCE bit in CR4 before the bus-cycle error, the processor also generates a machine-check exception.



Symbol	Description	Bits
LOCK	Locked Cycle	4
M/IO	Memory or I/O Cycle	3
D/C	Data or Code Cycle	2
W/R	Write or Read Cycle	1
CHK	Valid Machine-Check Data	0

Figure 3. Machine-Check Type Register (MCTR)

Table 8. Machine-Check Type Register (MCTR) Fields

Bit	Mnemonic	Description	Function
4	LOCK	Locked Cycle	Set to 1 if the processor was asserting LOCK during the bus cycle.
3	M/IO	Memory or I/O	1 = memory cycle, 0 = I/O cycle
2	D/C	Data or Code	1 = data cycle, 0 = code cycle
1	W/R	Write or Read	1 = write cycle, 0 = read cycle
0	CHK	Valid Machine-Check Data	The processor sets the CHK bit to 1 when both the MCTR and MCAR registers contain valid information. The processor clears the CHK bit to 0 when software reads the MCTR with the RDMSR instruction.

Time Stamp Counter (TSC)

With each processor clock cycle, the processor increments a 64-bit time stamp counter (TSC) MSR. The counter can be written or read using the WRMSR or RDMSR instructions when the ECX register contains the value 10h and CPL = 0. The counter can also be read using the RDTSC instruction, but the required privilege level for this instruction is determined by the Time Stamp Disable (TSD) bit in CR4. With either of these instructions, the EDX and EAX registers hold the upper and lower doublewords (dwords) of the 64-bit value to be written to or read from the TSC, as follows:

- *EDX*—Upper 32 bits of TSC
- *EAX*—Lower 32 bits of TSC

The TSC can be loaded with any arbitrary value. This feature is compatible with the Pentium processor.

Array Access Register (AAR)

The Array Access Register (AAR) contains pointers for testing the tag and data arrays for the instruction cache, data cache, 4-Kbyte TLB, and 4-Mbyte TLB. The AAR can be written or read with the WRMSR or RDMSR instruction when the ECX register contains the value 82h. Figure 4 shows the format of AAR.

The internal cache for the AMD-K5 processor is divided into two caches—a 16-Kbyte, 4-way, set-associative instruction cache and an 8-Kbyte, 4-way, set-associative data cache. Cache and TLB testing is often done by the BIOS or operating system during power-up.

Note: *The AMD-K6 processor does not contain this register. The AMD-K6 contains a built-in self-test (BIST) for all internal memories.*

The individual locations of all SRAM arrays on the AMD-K5 processor are accessible with the RDMSR and WRMSR instructions. To access an array location, set up the Array Access MSR code (82h) in ECX, and the array pointer in EDX. EAX holds the data to be read or written. For more information, see “Array Pointer.”

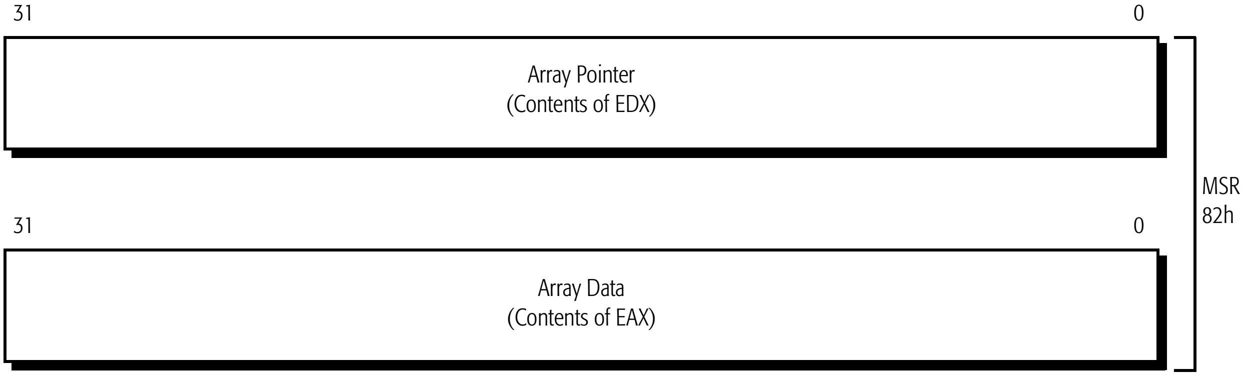


Figure 4. Array Access Register (AAR)

To read or write an array location, perform the following steps:

1. *ECX*—Enter 82h into ECX to access the 64-bit AAR.
2. *EDX*—Enter a 32-bit *array pointer* into EDX (the possible values are listed in Table 9, “Array IDs in Array Pointers,” on page 15.
3. *EAX*—Read or write 32 bits of *array test data* to or from EAX.

Array Pointer

The array pointers entered in EDX specify particular array locations. For example, in the data-cache and instruction-cache arrays, the way (or column) and set (or index) in the array pointer specify a cache line in the 4-way, set-associative array. The array pointers for data-cache data and instruction-cache instructions also specify a dword location within that cache line. In the data cache, this dword is 32 bits of data; in the instruction cache, this dword is two instruction bytes plus their associated pre-decode bits. For the 4-Kbyte TLB, the way and set specify one of the 128 TLB entries. In 4-Mbyte TLB, one of only four entries is specified.

Bits 7–0 of every array pointer encode the *array ID*, which identifies the array to be accessed, as shown in Table 9. To simplify multiple accesses to an array, the contents of EDX are retained after the RDMSR instruction executes (EDX is normally cleared after a RDMSR instruction).

Table 9. Array IDs in Array Pointers

Array Pointer Bits 7–0	Accessed Array
E0h	Data Cache: Data
E1h	Data Cache: Linear Tag
ECh	Data Cache: Physical Tag
E4h	Instruction Cache: Instructions
E5h	Instruction Cache: Linear Tag
EDh	Instruction Cache: Physical Tag
E6h	Instruction Cache: Valid Bits
E7h	Instruction Cache: Branch-Prediction Bits
E8h	4-Kbyte TLB: Page
E9h	4-Kbyte TLB: Virtual Tag
EAh	4-Mbyte TLB: Page
EBh	4-Mbyte TLB: Virtual Tag

Array Test Data

EAX specifies the test data to be read or written with the RDMSR or WRMSR instruction. For example, the array pointer in EDX specifies a way and set within the data-cache linear tag array (E1h in bits 7–0 of the array pointer) or the physical tag array (ECh in bits 7–0 of the array pointer). If the linear tag array (E1h) is accessed, the data read or written includes the tag and the status bits. The details of the valid fields in EAX are proprietary.

Hardware Configuration Register (HWCR)

The Hardware Configuration Register (HWCR) is a MSR that contains configuration bits that enable cache, branch tracing, write allocation, debug, and clock control functions. The WRMSR and RDMSR instructions access the HWCR when the ECX register contains the value 83h. Figure 5 and Table 10 show the format and fields of the HWCR.

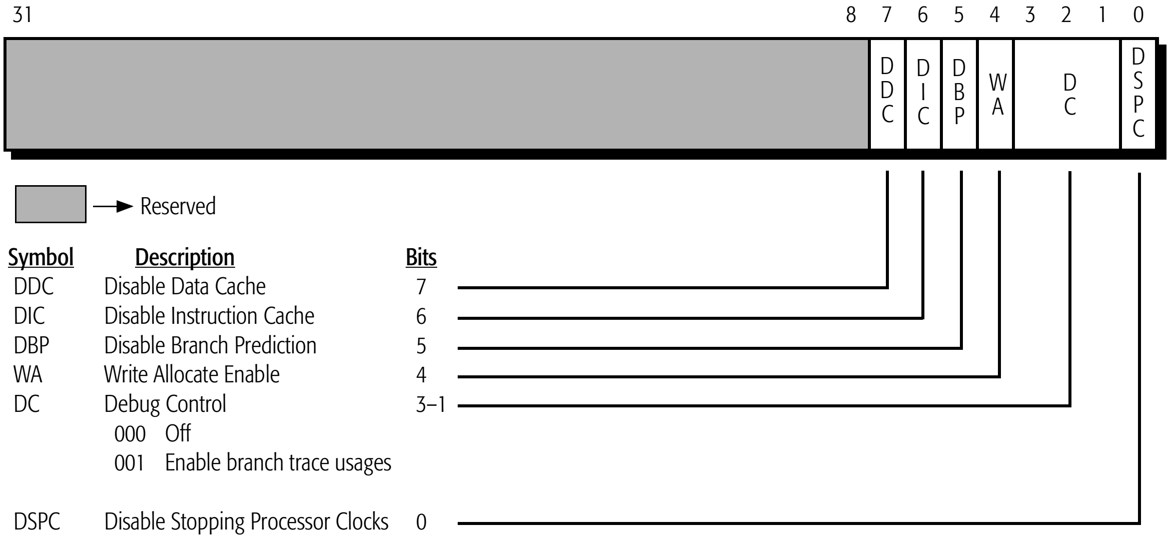


Figure 5. Hardware Configuration Register (HWCR)

Table 10. Hardware Configuration Register (HWCR) Fields

Bit	Mnemonic	Description	Function
31–8	–	–	*reserved*
7	DDC	Disable Data Cache	Disables data cache 0 = enabled, 1 = disabled
6	DIC	Disable Instruction Cache	Disables instruction cache 0 = enabled, 1 = disabled
5	DBP	Disable Branch Prediction	Disables branch prediction 0 = enabled, 1 = disabled
4	WA*	Enable Write Allocate	Enables write allocation 0 = disabled, 1 = enabled

Note:

* *The AMD-K5 processor supports write allocate only on Models 1, 2, and 3, with a Stepping of 4 or greater.*

Table 10. Hardware Configuration Register (HWCR) Fields (continued)

Bit	Mnemonic	Description	Function
3–1	DC	Debug Control	Debug control bits: 000 Off (disable HWCR debug control) 001 Enable branch-tracing messages 010 *reserved* 011 *reserved* 100 *reserved* 101 *reserved* 110 *reserved* 111 *reserved*
0	DSPC	Disable Stopping Processor Clocks	Disables stopping of internal processor clocks in the Halt and Stop Grant states 0 = enabled, 1 = disabled

Note:
* *The AMD-K5 processor supports write allocate only on Models 1, 2, and 3, with a Stepping of 4 or greater.*

Write Allocate Registers

A write allocate, if enabled, occurs when the processor has a pending memory write cycle to a cacheable line and the line does not currently reside in the L1 cache. The AMD-K5 processor supports write allocate only on Models 1, 2, and 3, with a Stepping of 4 or greater. Use the CPUID instruction to determine if the proper revision of the processor is present (See the *AMD Processor Recognition Application Note*, order# 20734, located at http://www.amd.com). For more information on implementing write allocate, see the *Implementation of Write Allocate in the K86™ Processors Application Note*, order# 21326.

The AMD-K5 processor implements write allocate by providing a global write allocate enable bit, three range-protection enable bits, and two memory range registers. The global write allocate enable bit is accessed using the Hardware Configuration Register (HWCR). The memory range registers and range enable bits are programmed by read/write MSR instructions.

Two MSRs are defined to support write allocate. The MSRs are accessed using the RDMSR and WRMSR instructions (See “RDMSR and WRMSR” in the *AMD K86 Family BIOS and Software Tools Developers Guide*, order# 21062).

The following index values in the ECX register access the MSRs:

- Write Allocate Top-of-Memory and Control Register (WATMCR)—ECX = 85h
- Write Allocate Programmable Memory Range Register (WAPMRR)—ECX = 86h

Three non-write-allocatable memory ranges are defined for use with the write allocate feature—one fixed range and two programmable ranges.

Fixed Range. The fixed memory range is 000A_0000h–000F_FFFFh and can be enabled or disabled. When enabled, write allocate can not be performed in this range.

Programmable Range. One programmable memory range is xxxx_0000h–yyyy_FFFFh, where xxxx and yyyy are defined using bits 15–0 and bits 31–16 of WAPMRR, respectively. Set bit 17 of WATMCR to enable protection of this range. When enabled, write allocate can not be performed in this range.

Top of Memory. The other programmable memory range is defined by the ‘top-of-memory’ field. The top of memory is equal to zzzz_0000h, where zzzz is defined using bits 15–0 of WATMCR. Addresses above zzzz_0000h are protected from write allocate when bit 18 of WATMCR is enabled.

Bits 18–16 of WATMCR control the enabling or disabling of the three memory ranges as follows:

- Bit 18: Top-of-Memory Enable bit

  0 = disabled (default)
  1 = enabled (write allocate can not be performed above Top of Memory)

- Bit 17: Programmable Range Enable bit

  0 = disabled (default)

  1 = enabled (write allocate can not be performed in this range)

- Bit 16: Fixed Range Enable bit

  0 = disabled (default)

  1 = enabled (write allocate can not be performed in this range)

Figures 6 and 7 show the bit positions for these two new registers.

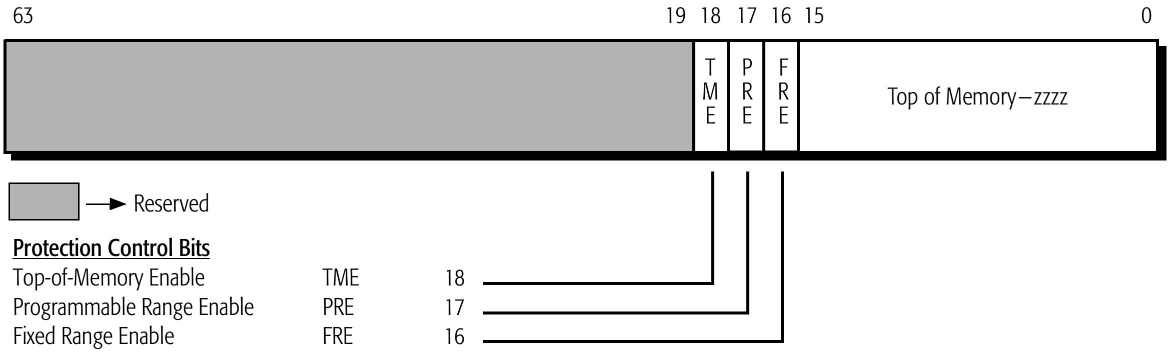

63
19
18
17
16
15
0
TME
PRE
FRE
Top of Memory—zzzz
Reserved
Protection Control Bits
Top-of-Memory Enable TME 18
Programmable Range Enable PRE 17
Fixed Range Enable FRE 16

Figure 6. Write Allocate Top-of-Memory and Control Register (WATMCR)—MSR 85h

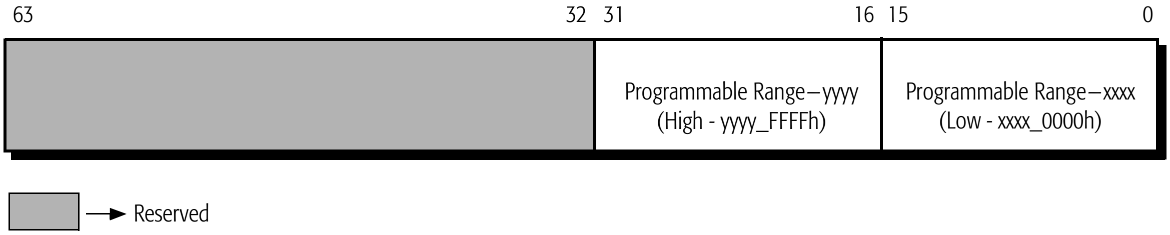


Figure 7. Write Allocate Programmable Memory Range Register (WAPMRR)—MSR 86h

AMD-K6® Processor

The AMD-K6 processor provides the following six MSRs. The contents of ECX selects the MSR to be addressed by the RDMSR and WRMSR instruction.

- Machine-Check Address Register (MCAR)—ECX = 00h
- Machine-Check Type Register (MCTR)—ECX = 01h
- Test Register 12 (TR12)—ECX = 0Eh
- Time Stamp Counter (TSC)—ECX = 10h
- Extended Feature Enable Register (EFER)—ECX = C000_0080h
- SYSCALL Target Address Register (STAR)—ECX = C000_0081h
- Write Handling Control Register (WHCR)—ECX = C000_0082h

Machine-Check Address Register (MCAR) and Machine-Check Type Register (MCTR)

The AMD-K6 processor does not support the generation of a machine check exception, but does provide a 64-bit Machine Check Address Register (MCAR) and a 64-bit Machine Check Type Register (MCTR) for software compatibility. Because the processor does not support machine check exceptions, the contents of the MCAR and MCTR are only affected by the WRMSR instruction and by RESET being sampled asserted (where all bits in each register are reset to 0).

The processor also provides the Machine Check Exception (MCE) bit in Control Register 4 (CR4, bit 6) as a read-write bit. However, the state of this bit has no effect on the operation of the processor.

Test Register 12 (TR12)

The AMD-K6 processor provides the 64-bit Test Register 12 (TR12), but only the Cache Inhibit (CI) bit (bit 3 of TR12) is supported. All other bits in TR12 have no effect on the processor's operation. The I/O Trap Restart function (bit 9 of TR12) is always enabled on the AMD-K6.

Time Stamp Counter (TSC)

See "Time Stamp Counter (TSC)" on page 13 in the AMD-K5 model-specific register section.

Extended Feature Enable Register (EFER)

The Extended Feature Enable Register (EFER) contains the control bits that enable the extended features of the AMD-K6 processor. Figure 8 shows the format of the EFER register, and Table 11 defines the function of each bit of the EFER register. The EFER register is MSR C000_0080h.

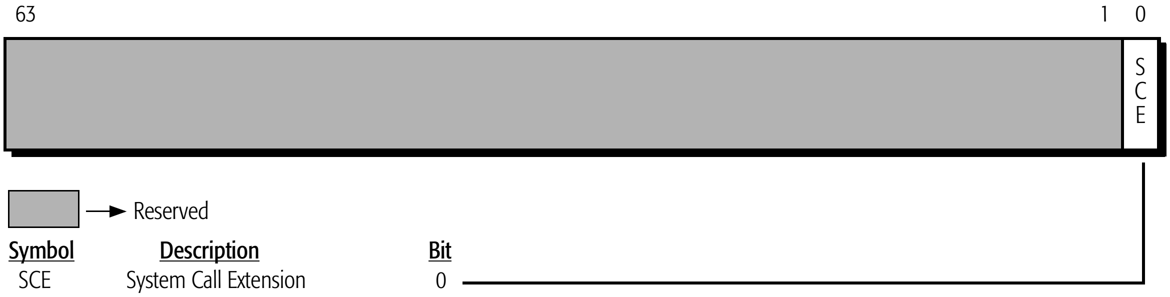


Figure 8. Extended Feature Enable Register (EFER)

Table 11. Extended Feature Enable Register (EFER) Definition

Bit	Description	R/W	Function
63–1	Reserved	R	Writing a 1 to any reserved bit causes a general protection fault to occur. All reserved bits are always read as 0.
0	System Call Extension (SCE)	R/W	SCE must be set to 1 to enable the usage of the SYSCALL and SYSRET instructions.

SYSCALL/SYSRET Target Address Register (STAR-MSR C000_0081h)

The SYSCALL/SYSRET Target Address Register (STAR) contains the target EIP address used by the SYSCALL instruction and the 16-bit code and stack segment selector bases used by the SYSCALL and SYSRET instructions. Figure 9 shows the format of the STAR register, and Table 12 on page 22 defines the function of each bit of the STAR register. For more information about SYSCALL/SYSRET, see the *AMD-K6® Processor SYSCALL and SYSRET Instruction Specification Application Note*, order# 21086.

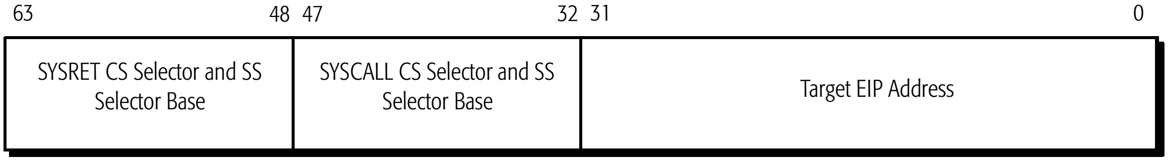

63
48
47
32
31
0
SYSRET CS Selector and SS Selector Base
SYSCALL CS Selector and SS Selector Base
Target EIP Address

Figure 9. SYSCALL/SYSRET Target Address Register (STAR)

Table 12. SYSCALL/SYSRET Target Address Register (STAR) Definition

Bit	Description	R/W	Function
63–48	SYSRET CS and SS Selector Base	R/W	During the SYSRET instruction, this field is copied into the CS register and the contents of this field, plus 1000b, are copied into the SS register.
47–32	SYSCALL CS and SS Selector Base	R/W	During the SYSCALL instruction, this field is copied into the CS register and the contents of this field, plus 1000b, are copied into the SS register.
31–0	Target EIP Address	R/W	This address is copied into the EIP and points to the new starting address.

Write Handling Control Register (WHCR)

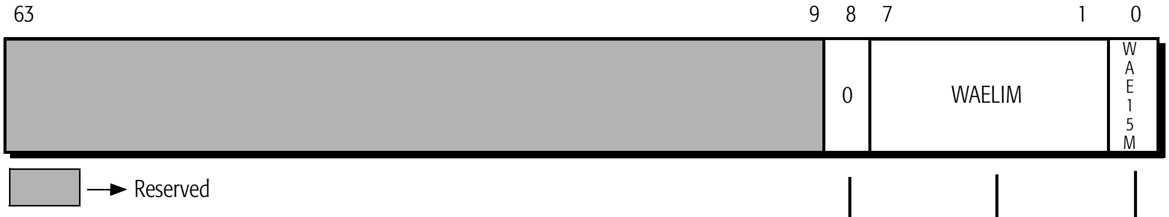
The AMD-K6 processor contains a split level-1 (L1) 64-Kbyte writeback cache organized as a separate 32-Kbyte instruction cache and a 32-Kbyte data cache with two-way set associativity. The cache line size is 32 bytes, and lines are read from memory using an efficient pipelined burst read cycle. Further performance gains are achieved by the implementation of a write allocation scheme.

A write allocate, if enabled, occurs when the processor has a pending memory write cycle to a cacheable line and the line does not currently reside in the L1 cache. For more information on write allocate, see the *Implementation of Write Allocate in the K86™ Processors Application Note*, order# 21326 and the Software Environment section of the *AMD-K6® Processor Data Sheet*, order# 20695.

The AMD-K6 processor implements write allocate differently than the AMD-K5 processor. This section describes two programmable mechanisms used by the AMD-K6 processor to determine when to perform write allocate. When either of these mechanisms indicates that a pending write is to a cacheable area of memory, a write allocate is performed.

Before programming the write allocate register or changing memory cacheability/writeability, the BIOS must writeback and invalidate the internal cache by using the WBINVD instruction. In addition, the write allocate register (WHCR) should enable the write allocate mechanisms only after performing any memory sizing or typing algorithms.

Write Handling Control Register (WHCR). The Write Handling Control Register (WHCR) is an MSR that contains three fields—the WCDE bit, the Write Allocate Enable Limit (WAELIM) field, and the Write Allocate Enable 15-to-16-Mbyte (WAE15M) bit (See Figure 10).



Symbol	Description	Bits
WCDE	Always program to 0	8
WAELIM	Write Allocate Enable Limit	7–1
WAE15M	Write Allocate Enable 15-to-16-Mbyte	0

***Note**: Hardware RESET initializes this MSR to all zeros.*

Figure 10. Write Handling Control Register (WHCR)—MSR C000_0082h

WCDE. For proper functionality, always program bit 8 to 0.

Write Allocate Enable Limit. The WAELIM field is 7 bits wide. This field, multiplied by 4 Mbytes, defines an upper memory limit. Any pending write cycle that addresses memory below this limit causes the processor to perform a write allocate. Write allocate is disabled for memory accesses at and above this limit unless the processor determines a pending write cycle is cacheable by means of one of the write allocate mechanisms—Write to a Cacheable Page and Write to a Sector (for more information, see the Cache chapter in the *AMD-K6® Processor Data Sheet*, order# 20695). The maximum value of this limit is $((2^7–1) \cdot 4$ Mbytes$)$ = 508 Mbytes. When all the bits in this field are set to 0, all memory is above this limit and the write allocate mechanism is disabled.

Write Allocate Enable 15-to-16-Mbyte. The WAE15M bit is used to enable write allocations for the memory write cycles that address the 1 Mbyte of memory between 15 Mbytes and 16 Mbytes. This bit must be set to 0 to prevent write allocates in this memory area. This sub-mechanism of the WAELIM provides a memory hole to prevent write allocates. This memory hole is provided to account for a small number of uncommon memory-mapped I/O adapters that use this particular memory address space. If the system contains one of these peripherals, the bit should be set to 0. The WAE15M bit is ignored if the value in the WAELIM field is set to less than 16 Mbytes.

X86 Processor Extensions

For more details on the new instructions, see the *AMD K86 Family BIOS and Software Tools Developers Guide*, order# 21062.

New AMD-K5™ Processor Instructions

In addition to supporting all 486 processor instructions, the AMD-K5 processor implements the following instructions:

- CPUID
- CMPXCHG8B
- MOV to and from CR4
- RDTSC
- RDMSR
- WRMSR
- RSM
- Illegal instruction (reserved opcode)

New AMD-K6® Processor Instructions

In addition to the new instructions implemented on the AMD-K5 processor, the AMD-K6 processor also implements the following instructions:

- SYSCALL
- SYSRET
- MMX—57 new instructions for multimedia software
- AMD-3D™—24 new instructions for multimedia software (Models 8 and above)

Additional Considerations

Software Timing Dependencies Relative to Memory Controller Setup

Processors in the K86 family differ from other processors with regards to instruction latencies and the order or priority of processor bus cycles. Timing-dependent software that relies on the specific latencies of other processors should be re-tested for proper operation with the K86 processor. In addition, re-testing should be performed on components with variable timing (such as, memory modules, oscillators, and timers).

Particular attention should be paid to memory-setup subroutines that determine the type of DRAM in the system. Some chipsets may not tolerate a DRAM mode change (such as, EDO to SDRAM) on the same clock as a DRAM refresh cycle. For example some chipsets do not tolerate having its memory refresh enabled prior to changing memory mode types. Refresh should only be enabled after the memory type has been determined.

Note: *The BIOS for the K86 family of processors should enable the write allocate mechanisms only after performing any memory sizing or typing algorithms.*

For details about supporting the AMD-640™ chipset, see the *AMD-640™ Chipset BIOS Design Application Note*, order# 21338.