

Intel® 64 and IA-32 Architectures Software Developer's Manual

Volume 4: Model-Specific Registers

NOTE: The Intel® 64 and IA-32 Architectures Software Developer's Manual consists of ten volumes: Basic Architecture, Order Number 253665; Instruction Set Reference, A-L, Order Number 253666; Instruction Set Reference, W-U, Order Number 253667; Instruction Set Reference, V, Order Number 326018; Instruction Set Reference, W-Z, Order Number 334569; System Programming Guide, Part 1, Order Number 253668; System Programming Guide, Part 2, Order Number 253669; System Programming Guide, Part 3, Order Number 326019; System Programming Guide, Part 4, Order Number 332831; Model-Specific Registers, Order Number 335592. Refer to all ten volumes when evaluating your design needs.

Order Number: 335592-078US

December 2022

Notices & Disclaimers

Intel technologies may require enabled hardware, software or service activation.

No product or component can be absolutely secure.

Your costs and results may vary.

You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.

All product plans and roadmaps are subject to change without notice.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

Code names are used by Intel to identify products, technologies, or services that are in development and not publicly available. These are not "commercial" names and not intended to function as trademarks.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document, with the sole exception that a) you may publish an unmodified copy and b) code included in this document is licensed subject to the Zero-Clause BSD open source license (0BSD), https://opensource.org/licenses/0BSD. You may create software implementations based on this document and in compliance with the foregoing that are intended to execute on the Intel product(s) referenced in this document. No rights are granted to create modifications or derivatives of this document.

© Intel Corporation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Other names and brands may be claimed as the property of others.

PAGE

CHAPT	TER 1 T THIS MANUAL	
	INTEL® 64 AND IA-32 PROCESSORS COVERED IN THIS MANUAL	
	OVERVIEW OF THE SYSTEM PROGRAMMING GUIDE	
	NOTATIONAL CONVENTIONS	
1.3.1	Bit and Byte Order	1-5
1.3.2	Reserved Bits and Software Compatibility	
1.3.3 1.3.4	Instruction Operands.	I-b
	Hexadecimal and Binary Numbers	1 6
1.3.5	Segmented Addressing	1-0
1.3.6 1.3.7	Syntax for CPUID, CR, and MSR Values	1-/
1.5.7	Exceptions	1.0
1.4	RELATED CITERATURE	1-0
CHAPT		
MODE	L-SPECIFIC REGISTERS (MSRS)	
2.1	ARCHITECTURAL MSRS	2-3
2.2	ARCHITECTURAL MSRS	2-67
2.3	MSRS IN THE 45 NM AND 32 NM INTEL ATOM® PROCESSOR FAMILY	2-81
2.4	MSRS IN INTEL PROCESSORS BASED ON SILVERMONT MICROARCHITECTURE	
2.4.1	MSRs with Model-Specific Behavior in the Silvermont Microarchitecture	2-105
2.4.2	MSRs In Intel Atom [®] Processors Based on Airmont Microarchitecture	2-109
	MSRS IN INTEL ATOM® PROCESSORS BASED ON GOLDMONT MICROARCHITECTURE	
2.6	MSRS IN INTEL ATOM® PROCESSORS BASED ON GOLDMONT PLUS MICROARCHITECTURE	2-135
2.7	MSRS IN INTEL ATOM® PROCESSORS BASED ON TREMONT MICROARCHITECTURE	2-139
2.8	MSRS IN THE NEHALEM MICROARCHITECTURE	2-141
2.8.1	MSRS IN THE NEHALEM MICROARCHITECTURE. Additional MSRs in the Intel® Xeon® Processor 5500 and 3400 Series. Additional MSRs in the Intel® Xeon® Processor 7500 Series. MSRS IN THE INTEL® XEON® PROCESSOR 5600 SERIES (BASED ON WESTMERE MICROARCHITECTURE) MSRS IN THE INTEL® XEON® PROCESSOR FAMILY (BASED ON WESTMERE MICROARCHITECTURE). MSRS IN INTEL® PROCESSOR FAMILY BASED ON SANDY BRIDGE MICROARCHITECTURE).	2-159
2.8.2	Additional MSRs in the Intel® Xeon® Processor 7500 Series	2-161
2.9	MSRS IN THE INTEL® XEON® PROCESSOR 5600 SERIES (BASED ON WESTMERE MICROARCHITECTURE)	2-175
2.10	MSRS IN THE INTEL® XEON® PROCESSOR E7 FAMILY (BASED ON WESTMERE MICROARCHITECTURE)	2-176
2.11	MSRS IN INTEL® PROCESSOR FAMILY BASED ON SANDY BRIDGE MICROARCHITECTURE	2-178
2.11.1	MSRS IN THE ZNO GENERATION INTEL [®] Core [®] Processor Family Based on Sandy Bridde Microarchitecture	Z-198
2.11.2	MSRs in the Intel [®] Xeon [®] Processor E5 Family Based on Sandy Bridge Microarchitecture	2-202
2.11.3	Additional Uncore PMU MSRs in the Intel® TAXON® Processor E5 Family	2-206
2.12	MSRS IN THE 3RD GENERATION INTEL® CORE PROCESSOR FAMILY (BASED ON IVY BRIDGE MICROARCHITECTURE) MSRs In Intel® Xeon® Processor E5 v2 Product Family (Based on Ivy Bridge-E Microarchitecture)	2-209
2.12.1	MSKs in Intel® Xeon® Processor E5 V2 Product Family (Based on IV) Bridge-E Microarchitecture)	2-213
2.12.2	Additional MSRs Supported by Intel® Xeon® Processor E/ Vz Family	2-220
2.12.3	Additional Uncore PMU MSRs in the Intel [®] Xeon [®] Processor E5 v2 and E7 v2 Families	2-222
2.13	MSRs in 4th Generation Intel® Core™ Processor Family (based on Haswell Microarchitecture)	2-225
2.13.1 2.13.2	Marks in 4th defined from their Cone Processor Painting (based on maswell file odd chitecture)	2 242
2.13.2	Additional residency Pisks supported in 4th define door in the Core Processors	2-243
2.14	Additional Residency MSRs Supported in 4th Generation Intel® Core™ Processors. MSRS IN INTEL® XEON® PROCESSOR E5 V3 AND E7 V3 PRODUCT FAMILY Additional Uncore PMU MSRs in the Intel® Xeon® Processor E5 v3 Family. MSRS IN INTEL® CORE™ M PROCESSORS AND 5TH GENERATION INTEL CORE PROCESSORS. MSRS IN INTEL® XEON® PROCESSORS E5 V4 FAMILY. Additional MSRS Supported in the Intel® Xeon® Processor D Product Family.	2-244 2-244
2.14.1	Additional office Lifth Light in the litter Agon - Life State 1 to response to the Lifth Agon - Additional office Lifth Light Intel Code Deducted Code and Ethickholm Intel Code Deducted Code	2 261
2.15	MSDS IN INITE! ® VEAN® PROCESSORS AND STH GENERATION INTEL CORE PROCESSORS	2 260
2.16.1	Additional MSPs Supported in the Intel® Years Processes D Broduct Earling	2 2 2 2 0
2.16.1	Additional MSRs Supported in Intel® Xeon® Processors E5 v4 and E7 v4 Families	2 200
2.17	MSRS IN THE 6TH GENERATION, 7TH GENERATION, 8TH GENERATION, 9TH GENERATION, 10TH GENERATION, 11TH	2-200
2.17	GENERATION, 12TH GENERATION, AND 13TH GENERATION INTEL® CORE PROCESSORS, INTEL® XEON® SCALABLE	
	PROCESSOR FAMILY, 2ND, 3RD, AND 4TH GENERATION INTEL® XEON® SCALABLE PROCESSOR FAMILY, 8TH GENERAT	ION
	INITES CODE 12 DEDICESCORS AND INITES YEARS EDUCESCORS	2-28/L
2.17.1	INTEL® CORE™ I3 PROCESSORS, AND INTEL® XEON® E PROCESSORS	U+
L.1/.1	Microarchitecture and Coffee Lake Microarchitecture	2-307
2.17.2	MSRs Specific to 8th Generation Intel® Core™ i3 Processors	
2.17.2	MSRs Introduced in 10th Generation Intel® Core™ Processors	2-315
2.17.3	MSRs Introduced in 10th Generation Intel® Core™ Processors based on Tiger Lake Microarchitecture	2-319
2.17.5	MSRs Introduced in 12th and 13th Generation Intel® Core™ Processors Supporting Performance Hybrid	
2476	Architecture	2-322
2.17.6	MSRs Introduced in Intel® Xeon® Scalable Processor Family	2-330

		PAGE
2.17.7	MSRs Specific to 3rd Generation Intel® Xeon® Scalable Processor Family based on Ice Lake Microarchitecture	22-341
2.17.8	MSRs Specific to 4th Generation Intel® Xeon® Scalable Processor Family Based on Sapphire Rapids	
	Microarchitecture	
2.18	Microarchitecture	
	7215/7285/7295 SERI <u>E</u> S	2-353
2.19	7215/7285/7295 SERIES®®	2-369
2.19.1	MSRs Unique to Intel® Xeon® Processor MP with L3 Cache	2-395
2.20	MSRS IN INTEL® CORE "SOLO AND INTEL® CORE "DUO PROCESSORS	2-396
2.21	MSRS IN THE PENTIUM M PROCESSOR	2-405
2.22	MSRS IN THE P6 FAMILY PROCESSORS.	2-412
2.23	MSRS IN PENTIUM PROCESSORS	2-421
2.24	MSR INDEX	2-422

PAGE

FIGURES

Figure 1-1.	Bit and Byte Order	1-5	,
Figure 1-2.	Syntax for CPUID, CR, and MSR Data Presentation	1-7	

TABLES

Table 2-1.	CPUID Signature Values of DisplayFamily_DisplayModel	2-1
Table 2-2.	IA.32 Architectural MSDs	2-3
Table 2-2.	IA-32 Architectural MSRs	2 67
Table 2-4.	MSRs in 45 nm and 32 nm Intel Atom® Processor Family	2-81
Table 2-5.	MSRs Supported by Intel Atom® Processors with CPUID Signature 06_27H	2-91
Table 2-6.	MSRs Common to the Silvermont Microarchitecture and Newer Microarchitectures for Intel Atom® Processors	
Table 2-7.	MSRs Common to the Silvermont and Airmont Microarchitectures	.2-101
Table 2-8.	Specific MSRs Supported by Intel Atom® Processors with CPUID Signatures 06_37H, 06_4AH, 06_5AH,	
	06 5DH	.2-106
Table 2-9.	Specific MSRs Supported by Intel Atom® Processor E3000 Series with CPUID Signature 06_37H	.2-107
Table 2-10.	Specific MSRs Supported by Intel Atom® Processor C2000 Series with CPUID Signature 06_4DH	
Table 2-11.	MSRs in Intel Atom® Processors Based on the Airmont Microarchitecture	
Table 2-12.	MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture	
	MSRs in Intel Atom Processors Based on the Goldmont Plus Microarchitecture	
Table 2-13. Table 2-14.		
	MSRs in Intel Atom® Processors Based on the Tremont Microarchitecture	
Table 2-15.	MSRs in Processors Based on Nehalem Microarchitecture	
Table 2-16.	Additional MSRs in Intel® Xeon® Processor 5500 and 3400 Series	
Table 2-17.	Additional MSRs in Intel® Xeon® Processor 7500 Series	
Table 2-18.	Additional MSRs Supported by Intel Processors Based on Westmere Microarchitecture	
Table 2-19.	Additional MSRs Supported by Intel® Xeon® Processor E7 Family	.2-176
Table 2-20.	MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture	
Table 2-21.	MSRs Supported by the 2nd Generation Intel® Core™ Processors (Sandy Bridge Microarchitecture)	2-198
Table 2-22.	Uncore PMU MSRs Supported by 2nd Generation Intel® Core™ Processors	2-200
Table 2-22.	Selected MSRs Supported by Intel® Xeon® Processors E5 Family Based on Sandy Bridge Microarchitecture	2 202
	Uncore PMU MSRs in Intel® Xeon® Processor E5 Family	2 206
Table 2-24.		
Table 2-25.	Additional MSRs Supported by 3rd Generation Intel® Core™ Processors (Based on Ivy Bridge Microarchitecture)	.2-209
Table 2-26.	MSRs Supported by Intel® Xeon® Processors E5 v2 Product Family (based on Ivy Bridge-E microarchitecture)	
Table 2-27.	Additional MSRs Supported by Intel® Xeon® Processor E7 v2 Family with DisplayFamily_DisplayModel Signature	
	06_3EH	.2-220
Table 2-28.	Uncore PMU MSRs in Intel® Xeon® Processor E5 v2 and E7 v2 Families	.2-222
Table 2-29.	Additional MSRs Supported by Processors based on the Haswell or Haswell-E microarchitectures	.2-225
Table 2-30.	MSRs Supported by 4th Generation Intel® Core™ Processors (Haswell Microarchitecture)	
Table 2-31.	Additional Residency MSRs Supported by 4th Generation Intel® Core™ Processors with DisplayFamily_DisplayM	
10010 2 311	Signature 06_45H	
Table 2-32.	Additional MSRs Supported by Intel® Xeon® Processor E5 v3 Family	2 245
Table 2-32.		
	Uncore PMU MSRs in Intel® Xeon® Processor E5 v3 Family	.2-255
Table 2-34.	Additional MSRs Common to Processors Based the Broadwell Microarchitectures	
Table 2-35.	Additional MSRs Supported by Intel® Core™ M Processors and 5th Generation Intel® Core™ Processors	.2-26/
Table 2-36.	Additional MSRs Common to Intel® Xeon® Processor D and Intel Xeon Processors E5 v4 Family Based on the	
	Broadwell Microarchitecture	
Table 2-37.	Additional MSRs Supported by Intel® Xeon® Processor D with DisplayFamily_DisplayModel 06_56H	.2-279
Table 2-38.	Additional MSRs Supported by Intel® Xeon® Processors with DisplayFamily_DisplayModel 06_4FH	.2-281
Table 2-39.	Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation,	
	10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors,	
	Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family,	
	8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors	2 205
T-N- 2 40	OUI DEFINITION OF THE STREET OF THE STREET OF THE CONTRACT OF THE STREET	.2-203
Table 2-40.	Uncore PMU MSRs Supported by 6th Generation, 7th Generation, and 8th Generation Intel® Core™ Processors,	2 205
	and 8th generation Intel® Core™ i3 Processors	.2-305
Table 2-41.	Additional MSRs Supported by 7th Generation and 8th Generation Intel® Core™ Processors Based on Kaby Lake	
	Microarchitecture and Coffee Lake Microarchitecture	.2-307
Table 2-42.	Additional MSRs Supported by 8th Generation Intel® Core™ i3 Processors Based on Cannon Lake	
	Microarchitecture	.2-309
Table 2-43.	Uncore PMU MSRs Supported by Intel® Core™ Processors Based on Cannon Lake Microarchitecture	
Table 2-44.	MSRs Supported by 10th Generation Intel® Core™ Processors Based on Ice Lake Microarchitecture	
Table 2-45.	Additional MSRs Supported by 11th Generation Intel® Core™ Processors Based on Tiger Lake	515
i avie 2-45.		2 210
T-LI- 2 40	Microarchitecture	.2-319 :-
Table 2-46.	Additional MSRs Supported by 12th and 13th Generation Intel® Core™ Processors Supporting Performance Hyb	
	Architecture	
Table 2-47.	MSRs Supported by 12th and 13th Generation Intel® Core™ Processor P-core	.2-326
Table 2-48.		
Tubic L To.	MSRs Supported by 12th and 13th Generation Intel® Core™ Processor E-core	.2-327
Table 2-49.	MSRs Supported by 12th and 13th Generation Intel® Core™ Processor E-core	.2-327 .2-327

PAGE

Table 2-51.	MSRs Supported by 3rd Generation Intel® Xeon® Scalable Processor Family with DisplayFamily_DisplayModel Signatures of O6 6AH and O6 6CH	2 2/1
Table 2-52.	Additional MSRs Supported by the 4th Generation Intel® Xeon® Scalable Processor Family with a	2-541
	DisplayFamily_DisplayModel Signature of 06_8FH	2-344
Table 2-53.	Selected MSRs Supported by Intel® Xeon Phi™ Processors with DisplayFamily_DisplayModel Signatures 06_5	
	and 06_85H	
Table 2-54.	Additional MSRs Supported by Intel® Xeon Phi™ Processor 7215, 7285, 7295 Series with DisplayFamily_Disp	layModel
	Signature 06_85H	2-368
Table 2-55.	MSRs in the Pentium® 4 and Intel® Xeon® Processors	2-369
Table 2-56.	MSRs Unique to 64-bit Intel® Xeon® Processor MP with Up to an 8 MB L3 Cache	2-395
Table 2-57.	MSRs Unique to Intel® Xeon® Processor 7100 Series	
Table 2-58.	MSRs in Intel® Core™ Solo, Intel® Core™ Duo Processors, and Dual-Core Intel® Xeon® Processor LV	2-396
Table 2-59.	MSRs in Pentium M Processors	2-405
Table 2-60.	MSRs in the P6 Family Processors	2-412
Table 2-61.	MSRs in the Pentium Processor	

CHAPTER 1 ABOUT THIS MANUAL

The Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 4: Model-Specific Registers (order number 335592) is part of a set that describes the architecture and programming environment of Intel® 64 and IA-32 architecture processors. Other volumes in this set are:

- Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 1: Basic Architecture (order number 253665).
- Intel® 64 and IA-32 Architectures Software Developer's Manual, Volumes 2A, 2B, 2C & 2D: Instruction Set Reference (order numbers 253666, 253667, 326018, and 334569).
- The Intel® 64 and IA-32 Architectures Software Developer's Manual, Volumes 3A, 3B, 3C & 3D: System Programming Guide (order numbers 253668, 253669, 326019, and 332831).

The Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 1, describes the basic architecture and programming environment of Intel 64 and IA-32 processors. The Intel® 64 and IA-32 Architectures Software Developer's Manual, Volumes 2A, 2B, 2C & 2D, describe the instruction set of the processor and the opcode structure. These volumes apply to application programmers and to programmers who write operating systems or executives. The Intel® 64 and IA-32 Architectures Software Developer's Manual, Volumes 3A, 3B, 3C & 3D, describe the operating-system support environment of Intel 64 and IA-32 processors. These volumes target operating-system and BIOS designers. In addition, Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 3B, and Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 3C address the programming environment for classes of software that host operating systems. The Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 4, describes the model-specific registers of Intel 64 and IA-32 processors.

1.1 INTEL® 64 AND IA-32 PROCESSORS COVERED IN THIS MANUAL

This manual set includes information pertaining primarily to the most recent Intel 64 and IA-32 processors, which include:

- Pentium[®] processors
- P6 family processors
- Pentium[®] 4 processors
- Pentium[®] M processors
- Intel[®] Xeon[®] processors
- Pentium[®] D processors
- Pentium[®] processor Extreme Editions
- 64-bit Intel[®] Xeon[®] processors
- Intel[®] Core[™] Duo processor
- Intel[®] Core[™] Solo processor
- Dual-Core Intel[®] Xeon[®] processor LV
- Intel[®] Core™2 Duo processor
- Intel[®] Core[™]2 Quad processor Q6000 series
- Intel[®] Xeon[®] processor 3000, 3200 series
- Intel[®] Xeon[®] processor 5000 series
- Intel[®] Xeon[®] processor 5100, 5300 series
- Intel[®] Core[™]2 Extreme processor X7000 and X6800 series
- Intel[®] Core[™]2 Extreme QX6000 series
- Intel[®] Xeon[®] processor 7100 series

ABOUT THIS MANUAL

- Intel[®] Pentium[®] Dual-Core processor
- Intel[®] Xeon[®] processor 7200, 7300 series
- Intel[®] Core[™]2 Extreme QX9000 series
- Intel[®] Xeon[®] processor 5200, 5400, 7400 series
- Intel[®] Core[™]2 Extreme processor QX9000 and X9000 series
- Intel[®] Core[™]2 Quad processor Q9000 series
- Intel[®] Core[™]2 Duo processor E8000, T9000 series
- Intel Atom[®] processors 200, 300, D400, D500, D2000, N200, N400, N2000, E2000, Z500, Z600, Z2000, C1000 series are built from 45 nm and 32 nm processes.
- Intel[®] Core[™] i7 processor
- Intel[®] Core[™] i5 processor
- Intel[®] Xeon[®] processor E7-8800/4800/2800 product families
- Intel[®] Core[™] i7-3930K processor
- 2nd generation Intel[®] Core[™] i7-2xxx, Intel[®] Core[™] i5-2xxx, Intel[®] Core[™] i3-2xxx processor series
- Intel[®] Xeon[®] processor E3-1200 product family
- Intel[®] Xeon[®] processor E5-2400/1400 product family
- Intel[®] Xeon[®] processor E5-4600/2600/1600 product family
- 3rd generation Intel[®] Core[™] processors
- Intel[®] Xeon[®] processor E3-1200 v2 product family
- Intel[®] Xeon[®] processor E5-2400/1400 v2 product families
- Intel[®] Xeon[®] processor E5-4600/2600/1600 v2 product families
- Intel[®] Xeon[®] processor E7-8800/4800/2800 v2 product families
- 4th generation Intel[®] Core[™] processors
- The Intel[®] Core[™] M processor family
- Intel[®] Core[™] i7-59xx Processor Extreme Edition
- Intel[®] Core[™] i7-49xx Processor Extreme Edition
- Intel[®] Xeon[®] processor E3-1200 v3 product family
- Intel[®] Xeon[®] processor E5-2600/1600 v3 product families
- 5th generation Intel[®] Core[™] processors
- Intel[®] Xeon[®] processor D-1500 product family
- Intel[®] Xeon[®] processor E5 v4 family
- Intel Atom[®] processor X7-Z8000 and X5-Z8000 series
- Intel Atom[®] processor Z3400 series
- Intel Atom[®] processor Z3500 series
- 6th generation Intel[®] Core[™] processors
- Intel[®] Xeon[®] processor E3-1500m v5 product family
- 7th generation Intel[®] Core[™] processors
- Intel[®] Xeon Phi[™] Processor 3200, 5200, 7200 Series
- Intel[®] Xeon[®] Scalable Processor Family
- 8th generation Intel[®] Core[™] processors
- Intel[®] Xeon Phi[™] Processor 7215, 7285, 7295 Series
- Intel[®] Xeon[®] E processors
- 9th generation Intel[®] Core[™] processors
- 2nd generation Intel[®] Xeon[®] Scalable Processor Family

- 10th generation Intel[®] Core[™] processors
- 11th generation Intel[®] Core[™] processors
- 3rd generation Intel[®] Xeon[®] Scalable Processor Family
- 12th generation Intel[®] Core[™] processors
- 13th generation Intel[®] Core[™] processors
- 4th generation Intel[®] Xeon[®] Scalable Processor Family

P6 family processors are IA-32 processors based on the P6 family microarchitecture. This includes the Pentium $^{\mathbb{R}}$ Pro, Pentium $^{\mathbb{R}}$ II, Pentium $^{\mathbb{R}}$ III, and Pentium $^{\mathbb{R}}$ III Xeon $^{\mathbb{R}}$ processors.

The Pentium $^{\mathbb{R}}$ 4, Pentium $^{\mathbb{R}}$ D, and Pentium $^{\mathbb{R}}$ processor Extreme Editions are based on the Intel NetBurst $^{\mathbb{R}}$ microarchitecture. Most early Intel $^{\mathbb{R}}$ Xeon $^{\mathbb{R}}$ processors are based on the Intel NetBurst $^{\mathbb{R}}$ microarchitecture. Intel Xeon processor 5000, 7100 series are based on the Intel NetBurst $^{\mathbb{R}}$ microarchitecture.

The Intel $^{\mathbb{R}}$ Core $^{\mathsf{TM}}$ Duo, Intel $^{\mathbb{R}}$ Core $^{\mathsf{TM}}$ Solo and dual-core Intel $^{\mathbb{R}}$ Xeon $^{\mathbb{R}}$ processor LV are based on an improved Pentium $^{\mathbb{R}}$ M processor microarchitecture.

The Intel[®] Xeon[®] processor 3000, 3200, 5100, 5300, 7200, and 7300 series, Intel[®] Pentium[®] dual-core, Intel[®] Core[™]2 Duo, Intel[®] Core[™]2 Quad, and Intel[®] Core[™]2 Extreme processors are based on Intel[®] Core[™] microarchitecture.

The Intel[®] Xeon[®] processor 5200, 5400, 7400 series, Intel[®] Core[™]2 Quad processor Q9000 series, and Intel[®] Core[™]2 Extreme processors QX9000, X9000 series, Intel[®] Core[™]2 processor E8000 series are based on Enhanced Intel[®] Core[™] microarchitecture.

The Intel Atom $^{\otimes}$ processors 200, 300, D400, D500, D2000, N200, N400, N2000, E2000, Z500, Z600, Z2000, C1000 series are based on the Intel Atom $^{\otimes}$ microarchitecture and supports Intel 64 architecture.

P6 family, Pentium[®] M, Intel[®] Core[™] Solo, Intel[®] Core[™] Duo processors, dual-core Intel[®] Xeon[®] processor LV, and early generations of Pentium 4 and Intel Xeon processors support IA-32 architecture. The Intel[®] Atom[™] processor Z5xx series support IA-32 architecture.

The Intel[®] Xeon[®] processor 3000, 3200, 5000, 5100, 5200, 5300, 5400, 7100, 7200, 7300, 7400 series, Intel[®] Core[™]2 Duo, Intel[®] Core[™]2 Extreme, Intel[®] Core[™]2 Quad processors, Pentium[®] D processors, Pentium[®] Dual-Core processor, newer generations of Pentium 4 and Intel Xeon processor family support Intel[®] 64 architecture.

The Intel[®] Core[™] i7 processor and Intel[®] Xeon[®] processor 3400, 5500, 7500 series are based on 45 nm Nehalem microarchitecture. Westmere microarchitecture is a 32 nm version of the Nehalem microarchitecture. Intel[®] Xeon[®] processor 5600 series, Intel Xeon processor E7 and various Intel Core i7, i5, i3 processors are based on the Westmere microarchitecture. These processors support Intel 64 architecture.

The Intel[®] Xeon[®] processor E5 family, Intel[®] Xeon[®] processor E3-1200 family, Intel[®] Xeon[®] processor E7-8800/4800/2800 product families, Intel[®] Core[™] i7-3930K processor, and 2nd generation Intel[®] Core[™] i7-2xxx, Intel[®] Core[™] i3-2xxx processor series are based on the Sandy Bridge microarchitecture and support Intel 64 architecture.

The Intel[®] Xeon[®] processor E7-8800/4800/2800 v2 product families, Intel[®] Xeon[®] processor E3-1200 v2 product family and 3rd generation Intel[®] Core[™] processors are based on the Ivy Bridge microarchitecture and support Intel 64 architecture.

The Intel[®] Xeon[®] processor E5-4600/2600/1600 v2 product families, Intel[®] Xeon[®] processor E5-2400/1400 v2 product families and Intel[®] Core^{IM} i7-49xx Processor Extreme Edition are based on the Ivy Bridge-E microarchitecture and support Intel 64 architecture.

The Intel $^{\$}$ Xeon $^{\$}$ processor E3-1200 v3 product family and 4th Generation Intel $^{\$}$ Core $^{\intercal}$ processors are based on the Haswell microarchitecture and support Intel 64 architecture.

The Intel[®] Xeon[®] processor E5-2600/1600 v3 product families and the Intel[®] Core[™] i7-59xx Processor Extreme Edition are based on the Haswell-E microarchitecture and support Intel 64 architecture.

The Intel Atom[®] processor Z8000 series is based on the Airmont microarchitecture.

The Intel Atom[®] processor Z3400 series and the Intel Atom[®] processor Z3500 series are based on the Silvermont microarchitecture.

The Intel[®] Core[™] M processor family, 5th generation Intel[®] Core[™] processors, Intel[®] Xeon[®] processor D-1500 product family and the Intel[®] Xeon[®] processor E5 v4 family are based on the Broadwell microarchitecture and support Intel 64 architecture.

The Intel[®] Xeon[®] Scalable Processor Family, Intel[®] Xeon[®] processor E3-1500m v5 product family and 6th generation Intel[®] Core[™] processors are based on the Skylake microarchitecture and support Intel 64 architecture.

The 7th generation Intel[®] Core[™] processors are based on the Kaby Lake microarchitecture and support Intel 64 architecture.

The Intel Atom[®] processor C series, the Intel Atom[®] processor X series, the Intel[®] Pentium[®] processor J series, the Intel[®] Celeron[®] processor J series, and the Intel[®] Celeron[®] processor N series are based on the Goldmont microarchitecture.

The Intel[®] Xeon Phi[™] Processor 3200, 5200, 7200 Series is based on the Knights Landing microarchitecture and supports Intel 64 architecture.

The Intel[®] Pentium[®] Silver processor series, the Intel[®] Celeron[®] processor J series, and the Intel[®] Celeron[®] processor N series are based on the Goldmont Plus microarchitecture.

The 8th generation Intel[®] Core[™] processors, 9th generation Intel[®] Core[™] processors, and Intel[®] Xeon[®] E processors are based on the Coffee Lake microarchitecture and support Intel 64 architecture.

The Intel[®] Xeon Phi[™] Processor 7215, 7285, 7295 Series is based on the Knights Mill microarchitecture and supports Intel 64 architecture.

The 2nd generation Intel[®] Xeon[®] Scalable Processor Family is based on the Cascade Lake product and supports Intel 64 architecture.

Some 10th generation $Intel^{\textcircled{R}}$ CoreTM processors are based on the Ice Lake microarchitecture, and some are based on the Comet Lake microarchitecture; both support Intel 64 architecture.

Some 11th generation Intel[®] CoreTM processors are based on the Tiger Lake microarchitecture, and some are based on the Rocket Lake microarchitecture; both support Intel 64 architecture.

Some 3rd generation Intel[®] Xeon[®] Scalable Processor Family processors are based on the Cooper Lake product, and some are based on the Ice Lake microarchitecture; both support Intel 64 architecture.

The 12th generation $Intel^{\textcircled{8}}$ $Core^{\intercal}$ processors are based on the Alder Lake performance hybrid architecture and support Intel 64 architecture.

The 13th generation Intel[®] CoreTM processors are based on the Raptor Lake performance hybrid architecture and support Intel 64 architecture.

The 4th generation Intel[®] Xeon[®] Scalable Processor Family is based on Sapphire Rapids microarchitecture and supports Intel 64 architecture.

IA-32 architecture is the instruction set architecture and programming environment for Intel's 32-bit microprocessors. Intel[®] 64 architecture is the instruction set architecture and programming environment which is the superset of Intel's 32-bit and 64-bit architectures. It is compatible with the IA-32 architecture.

1.2 OVERVIEW OF THE SYSTEM PROGRAMMING GUIDE

A description of this manual's content follows:

Chapter 1 — About This Manual. Gives an overview of all eight volumes of the *Intel*® *64 and IA-32 Architectures Software Developer's Manual.* It also describes the notational conventions in these manuals and lists related Intel manuals and documentation of interest to programmers and hardware designers.

Chapter 2 — Model-Specific Registers (MSRs). Lists the MSRs available in Intel processors, and describes their functions.

1.3 NOTATIONAL CONVENTIONS

This manual uses specific notation for data-structure formats, for symbolic representation of instructions, and for hexadecimal and binary numbers. A review of this notation makes the manual easier to read.

1.3.1 Bit and Byte Order

In illustrations of data structures in memory, smaller addresses appear toward the bottom of the figure; addresses increase toward the top. Bit positions are numbered from right to left. The numerical value of a set bit is equal to two raised to the power of the bit position. Intel 64 and IA-32 processors are "little endian" machines; this means the bytes of a word are numbered starting from the least significant byte. Figure 1-1 illustrates these conventions.

1.3.2 Reserved Bits and Software Compatibility

In many register and memory layout descriptions, certain bits are marked as **reserved**. When bits are marked as reserved, it is essential for compatibility with future processors that software treat these bits as having a future, though unknown, effect. The behavior of reserved bits should be regarded as not only undefined, but unpredictable. Software should follow these guidelines in dealing with reserved bits:

- Do not depend on the states of any reserved bits when testing the values of registers which contain such bits.
 Mask out the reserved bits before testing.
- Do not depend on the states of any reserved bits when storing to memory or to a register.
- Do not depend on the ability to retain information written into any reserved bits.
- When loading a register, always load the reserved bits with the values indicated in the documentation, if any, or reload them with values previously read from the same register.

NOTE

Avoid any software dependence upon the state of reserved bits in Intel 64 and IA-32 registers. Depending upon the values of reserved register bits will make software dependent upon the unspecified manner in which the processor handles these bits. Programs that depend upon reserved values risk incompatibility with future processors.

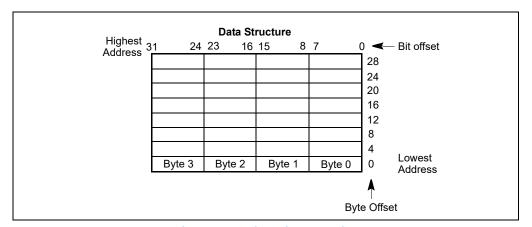


Figure 1-1. Bit and Byte Order

1.3.3 Instruction Operands

When instructions are represented symbolically, a subset of assembly language is used. In this subset, an instruction has the following format:

label: mnemonic argument1, argument2, argument3

where:

- A **label** is an identifier which is followed by a colon.
- A mnemonic is a reserved name for a class of instruction opcodes which have the same function.
- The operands **argument1**, **argument2**, and **argument3** are optional. There may be from zero to three operands, depending on the opcode. When present, they take the form of either literals or identifiers for data items. Operand identifiers are either reserved names of registers or are assumed to be assigned to data items declared in another part of the program (which may not be shown in the example).

When two operands are present in an arithmetic or logical instruction, the right operand is the source and the left operand is the destination.

For example:

LOADREG: MOV EAX, SUBTOTAL

In this example LOADREG is a label, MOV is the mnemonic identifier of an opcode, EAX is the destination operand, and SUBTOTAL is the source operand. Some assembly languages put the source and destination in reverse order.

1.3.4 Hexadecimal and Binary Numbers

Base 16 (hexadecimal) numbers are represented by a string of hexadecimal digits followed by the character H (for example, F82EH). A hexadecimal digit is a character from the following set: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, and F.

Base 2 (binary) numbers are represented by a string of 1s and 0s, sometimes followed by the character B (for example, 1010B). The "B" designation is only used in situations where confusion as to the type of number might arise.

1.3.5 Segmented Addressing

The processor uses byte addressing. This means memory is organized and accessed as a sequence of bytes. Whether one or more bytes are being accessed, a byte address is used to locate the byte or bytes memory. The range of memory that can be addressed is called an **address space**.

The processor also supports segmented addressing. This is a form of addressing where a program may have many independent address spaces, called **segments**. For example, a program can keep its code (instructions) and stack in separate segments. Code addresses would always refer to the code space, and stack addresses would always refer to the stack space. The following notation is used to specify a byte address within a segment:

Segment-register: Byte-address

For example, the following segment address identifies the byte at address FF79H in the segment pointed by the DS register:

DS:FF79H

The following segment address identifies an instruction address in the code segment. The CS register points to the code segment and the EIP register contains the address of the instruction.

CS:EIP

1.3.6 Syntax for CPUID, CR, and MSR Values

Obtain feature flags, status, and system information by using the CPUID instruction, by checking control register bits, and by reading model-specific registers. We are moving toward a single syntax to represent this type of information. See Figure 1-2.

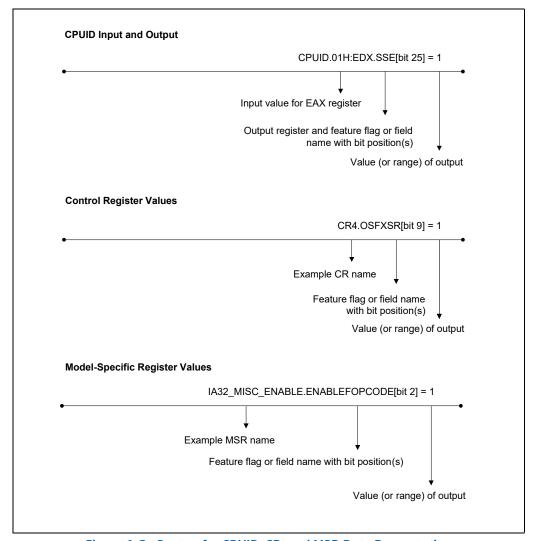


Figure 1-2. Syntax for CPUID, CR, and MSR Data Presentation

1.3.7 Exceptions

An exception is an event that typically occurs when an instruction causes an error. For example, an attempt to divide by zero generates an exception. However, some exceptions, such as breakpoints, occur under other conditions. Some types of exceptions may provide error codes. An error code reports additional information about the error. An example of the notation used to show an exception and error code is shown below:

#PF(fault code)

This example refers to a page-fault exception under conditions where an error code naming a type of fault is reported. Under some conditions, exceptions which produce error codes may not be able to report an accurate code. In this case, the error code is zero, as shown below for a general-protection exception:

#GP(0)

1.4 RELATED LITERATURE

Literature related to Intel 64 and IA-32 processors is listed and viewable on-line at:

https://software.intel.com/en-us/articles/intel-sdm

See also:

- The latest security information on Intel[®] products: https://www.intel.com/content/www/us/en/security-center/default.html
- Software developer resources, guidance, and insights for security advisories: https://software.intel.com/security-software-guidance/
- The data sheet for a particular Intel 64 or IA-32 processor
- The specification update for a particular Intel 64 or IA-32 processor
- Intel[®] C++ Compiler documentation and online help: http://software.intel.com/en-us/articles/intel-compilers/
- Intel[®] Fortran Compiler documentation and online help: http://software.intel.com/en-us/articles/intel-compilers/
- Intel[®] Software Development Tools: https://software.intel.com/en-us/intel-sdp-home
- Intel[®] 64 and IA-32 Architectures Software Developer's Manual (in one, four or ten volumes): https://software.intel.com/en-us/articles/intel-sdm
- Intel® 64 and IA-32 Architectures Optimization Reference Manual: https://software.intel.com/en-us/articles/intel-sdm#optimization
- Intel[®] Trusted Execution Technology Measured Launched Environment Programming Guide:

http://www.intel.com/content/www/us/en/software-developers/intel-txt-software-development-guide.html

- Intel[®] Software Guard Extensions (Intel[®] SGX) Information https://software.intel.com/en-us/isa-extensions/intel-sgx
- Developing Multi-threaded Applications: A Platform Consistent Approach: https://software.intel.com/sites/default/files/article/147714/51534-developing-multithreaded-applications.pdf
- Using Spin-Loops on Intel[®] Pentium[®] 4 Processor and Intel[®] Xeon[®] Processor: https://software.intel.com/sites/default/files/22/30/25602
- Performance Monitoring Unit Sharing Guide http://software.intel.com/file/30388

Literature related to select features in future Intel processors are available at:

 Intel[®] Architecture Instruction Set Extensions Programming Reference https://software.intel.com/en-us/isa-extensions

More relevant links are:

Intel[®] Developer Zone:

https://software.intel.com/en-us

Developer centers:

http://www.intel.com/content/www/us/en/hardware-developers/developer-centers.html

Processor support general link:

http://www.intel.com/support/processors/

Intel[®] Hyper-Threading Technology (Intel[®] HT Technology):

http://www.intel.com/technology/platform-technology/hyper-threading/index.htm

CHAPTER 2 MODEL-SPECIFIC REGISTERS (MSRS)

This chapter lists MSRs across Intel processor families. All MSRs listed can be read with the RDMSR and written with the WRMSR instructions. The scope of an MSR defines the set of processors that access the same MSR with RDMSR and WRMSR. Thread-scope MSRs are unique to every logical processor. Core-scope MSRs are shared by the threads in the same core; similarly for module-scope, die-scope, and package-scope.

When a processor package contains a single die, die-scope and package-scope are synonymous. When a package contains multiple die, they are distinct.

NOTE

For information on hierarchical level types supported, refer to the CPUID Leaf 1FH definition for the actual level type numbers: "V2 Extended Topology Enumeration Leaf" in the Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 2A. Also see Section 9.9.1, "Hierarchical Mapping of Shared Resources" in the Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 3A.

Register addresses are given in both hexadecimal and decimal. The register name is the mnemonic register name and the bit description describes individual bits in registers.

Model specific registers and its bit-fields may be supported for a finite range of processor families/models. To distinguish between different processor family and/or models, software must use CPUID.01H leaf function to query the combination of DisplayFamily and DisplayModel to determine model-specific availability of MSRs (see CPUID instruction in Chapter 3, "Instruction Set Reference, A-L" in the Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 2A). Table 2-1 lists the signature values of DisplayFamily and DisplayModel for various processor families or processor number series.

Table 2-1. CPUID Signature Values of DisplayFamily_DisplayModel

DisplayFamily_DisplayModel	Processor Families/Processor Number Series
06_85H	Intel® Xeon Phi™ Processor 7215, 7285, 7295 Series based on Knights Mill microarchitecture
06_57H	Intel® Xeon Phi™ Processor 3200, 5200, 7200 Series based on Knights Landing microarchitecture
06_8FH	4th generation Intel® Xeon® Scalable Processor Family based on Sapphire Rapids microarchitecture
06_BAH, 06_B7H, 06_BFH	13th generation Intel® Core™ processors supporting Raptor Lake performance hybrid architecture
06_97H, 06_9AH	12th generation Intel® Core™ processors supporting Alder Lake performance hybrid architecture
06_8CH, 06_8DH	11th generation Intel® Core™ processors based on Tiger Lake microarchitecture
06_A7H, 06_A8H	11th generation Intel® Core™ processors based on Rocket Lake microarchitecture
06_7DH, 06_7EH	10th generation Intel® Core™ processors based on Ice Lake microarchitecture
06_A5H, 06_A6H 10th generation Intel® Core™ processors based on Comet Lake microarchitecture	
06_66H Intel® Core™ processors based on Cannon Lake microarchitecture	
06_8ЕН, 06_9ЕН	7th generation Intel® Core™ processors based on Kaby Lake microarchitecture, 8th and 9th generation Intel® Core™ processors based on Coffee Lake microarchitecture, Intel® Xeon® E processors based on Coffee Lake microarchitecture
06_6AH, 06_6CH	3rd generation Intel® Xeon® Scalable Processor Family based on Ice Lake microarchitecture
06_55H	Intel® Xeon® Scalable Processor Family based on Skylake microarchitecture, 2nd generation Intel® Xeon® Scalable Processor Family based on Cascade Lake product, and 3rd generation Intel® Xeon® Scalable Processor Family based on Cooper Lake product
06_4EH, 06_5EH	6th generation Intel Core processors and Intel Xeon processor E3-1500m v5 product family and E3- 1200 v5 product family based on Skylake microarchitecture

Table 2-1. CPUID Signature (Contd.) Values of DisplayFamily_DisplayModel (Contd.)

	CPOID Signature (Conto.) values of DisplayFamily_DisplayModer (Conto.)		
DisplayFamily_DisplayModel			
06_56H	Intel Xeon processor D-1500 product family based on Broadwell microarchitecture		
06_4FH	Intel Xeon processor E5 v4 Family based on Broadwell microarchitecture, Intel Xeon processor E7 v4 Family, Intel Core i7-69xx Processor Extreme Edition		
06_47Н	5th generation Intel Core processors, Intel Xeon processor E3-1200 v4 product family based on Broadwell microarchitecture		
06_3DH	Intel Core M-5xxx Processor, 5th generation Intel Core processors based on Broadwell microarchitecture		
06_3FH	Intel Xeon processor E5-4600/2600/1600 v3 product families, Intel Xeon processor E7 v3 product families based on Haswell-E microarchitecture, Intel Core i7-59xx Processor Extreme Edition		
06_3CH, 06_45H, 06_46H	4th Generation Intel Core processor and Intel Xeon processor E3-1200 v3 product family based on Haswell microarchitecture		
06_3EH	Intel Xeon processor E7-8800/4800/2800 v2 product families based on Ivy Bridge-E microarchitecture		
06_3EH	Intel Xeon processor E5-2600/1600 v2 product families and Intel Xeon processor E5-2400 v2 product family based on Ivy Bridge-E microarchitecture, Intel Core i7-49xx Processor Extreme Edition		
06_3AH	3rd Generation Intel Core Processor and Intel Xeon processor E3-1200 v2 product family based on lvy Bridge microarchitecture		
06_2DH	Intel Xeon processor E5 Family based on Sandy Bridge microarchitecture, Intel Core i7-39xx Processor Extreme Edition		
06_2FH	Intel Xeon Processor E7 Family		
06_2AH	Intel Xeon processor E3-1200 product family; 2nd Generation Intel Core i7, i5, i3 Processors 2xxx Series		
06_2EH	Intel Xeon processor 7500, 6500 series		
06_25H, 06_2CH	Intel Xeon processors 3600, 5600 series, Intel Core i7, i5, and i3 Processors		
06_1EH, 06_1FH	Intel Core i7 and i5 Processors		
06_1AH	Intel Core i7 Processor, Intel Xeon processor 3400, 3500, 5500 series		
06_1DH	Intel Xeon processor MP 7400 series		
06_17H	Intel Xeon processor 3100, 3300, 5200, 5400 series, Intel Core 2 Quad processors 8000, 9000 series		
06_0FH	Intel Xeon processor 3000, 3200, 5100, 5300, 7300 series, Intel Core 2 Quad processor 6000 series, Intel Core 2 Extreme 6000 series, Intel Core 2 Duo 4000, 5000, 6000, 7000 series processors, Intel Pentium dual-core processors		
06_0EH	Intel Core Duo, Intel Core Solo processors		
06_0DH	Intel Pentium M processor		
06_86H, 06_96H, 06_9CH	Intel Atom® processors, Intel® Celeron® processors, Intel® Pentium® processors, and Intel® Pentium® Silver processors based on Tremont Microarchitecture		
06_7AH	Intel Atom processors based on Goldmont Plus microarchitecture		
06_5FH	Intel Atom processors based on Goldmont microarchitecture (Denverton)		
06_5CH	Intel Atom processors based on Goldmont microarchitecture		
06_4CH	Intel Atom processor X7-Z8000 and X5-Z8000 series based on Airmont microarchitecture		
06_5DH	Intel Atom processor X3-C3000 based on Silvermont microarchitecture		
06_5AH	Intel Atom processor Z3500 series		
06_4AH	Intel Atom processor Z3400 series		
06_37H	Intel Atom processor E3000 series, Z3600 series, Z3700 series		
06_4DH	Intel Atom processor C2000 series		

Table 2-1. CPUID Signature (Contd.) Values of DisplayFamily_DisplayModel (Contd.)

DisplayFamily_DisplayModel	Processor Families/Processor Number Series
06_36H	Intel Atom processor S1000 Series
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	Intel Atom processor family, Intel Atom processor D2000, N2000, E2000, Z2000, C1000 series
OF_06H	Intel Xeon processor 7100, 5000 Series, Intel Xeon Processor MP, Intel Pentium 4, Pentium D processors
0F_03H, 0F_04H	Intel Xeon processor, Intel Xeon processor MP, Intel Pentium 4, Pentium D processors
06_09H	Intel Pentium M processor
0F_02H	Intel Xeon Processor, Intel Xeon processor MP, Intel Pentium 4 processors
0F_0H, 0F_01H	Intel Xeon Processor, Intel Xeon processor MP, Intel Pentium 4 processors
06_7H, 06_08H, 06_0AH, 06_0BH	Intel Pentium III Xeon processor, Intel Pentium III processor
06_03H, 06_05H	Intel Pentium II Xeon processor, Intel Pentium II processor
06_01H	Intel Pentium Pro processor
05_01H, 05_02H, 05_04H	Intel Pentium processor, Intel Pentium processor with MMX Technology

The Intel® Quark $^{\mathbb{N}}$ SoC X1000 processor can be identified by the signature of DisplayFamily_DisplayModel = 05_09H and SteppingID = 0

2.1 ARCHITECTURAL MSRS

Many MSRs have carried over from one generation of IA-32 processors to the next and to Intel 64 processors. A subset of MSRs and associated bit fields, which do not change on future processor generations, are now considered architectural MSRs. For historical reasons (beginning with the Pentium 4 processor), these "architectural MSRs" were given the prefix "IA32_". Table 2-2 lists the architectural MSRs, their addresses, their current names, their names in previous IA-32 processors, and bit fields that are considered architectural. MSR addresses outside Table 2-2 and certain bit fields in an MSR address that may overlap with architectural MSR addresses are model-specific. Code that accesses a model-specific MSR and that is executed on a processor that does not support that MSR will generate an exception.

Architectural MSR or individual bit fields in an architectural MSR may be introduced or transitioned at the granularity of certain processor family/model or the presence of certain CPUID feature flags. The right-most column of Table 2-2 provides information on the introduction of each architectural MSR or its individual fields. This information is expressed either as signature values of "DF DM" (see Table 2-1) or via CPUID flags.

Certain bit field position may be related to the maximum physical address width, the value of which is expressed as "MAXPHYADDR" in Table 2-2. "MAXPHYADDR" is reported by CPUID.8000_0008H leaf.

MSR address range between 40000000H - 4000FFFFH is marked as a specially reserved range. All existing and future processors will not implement any features using any MSR in this range.

Table 2-2. IA-32 Architectural MSRs

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
ОН	0	IA32_P5_MC_ADDR (P5_MC_ADDR)	See Section 2.23, "MSRs in Pentium Processors."	Pentium Processor (05_01H)
1H	1	IA32_P5_MC_TYPE (P5_MC_TYPE)	See Section 2.23, "MSRs in Pentium Processors."	DF_DM = 05_01H
6H	6	IA32_MONITOR_FILTER_SIZE	See Section 9.10.5, "Monitor/Mwait Address Range Determination."	0F_03H

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
10H	16	IA32_TIME_STAMP_COUNTER (TSC)	See Section 18.17, "Time-Stamp Counter."	05_01H
17H	23	IA32_PLATFORM_ID (MSR_PLATFORM_ID)	Platform ID (R/O) The operating system can use this MSR to determine "slot" information for the processor and the proper microcode update to load.	06_01H
		49:0	Reserved	
1BH	27	63:53 IA32_APIC_BASE (APIC_BASE)	Platform Id (R/O) Contains information concerning the intended platform for the processor. 52 51 50 0 0 0 Processor Flag 0 0 0 1 Processor Flag 1 0 1 0 Processor Flag 2 0 1 1 Processor Flag 3 1 0 0 Processor Flag 4 1 0 1 Processor Flag 5 1 1 0 Processor Flag 6 1 1 1 Processor Flag 7 Reserved This register holds the APIC base address, permitting the relocation of the APIC	06_01H
			memory map. See Section 11.4.4, "Local APIC Status and Location" and Section 11.4.5, "Relocating the Local APIC Registers".	
		7:0	Reserved	
		8	BSP flag (R/W)	
		9	Reserved	
		10	Enable x2APIC mode.	06_1AH
		11	APIC Global Enable (R/W)	
		(MAXPHYADDR - 1):12	APIC Base (R/W)	
		63: MAXPHYADDR	Reserved	
ЗАН	58	IA32_FEATURE_CONTROL	Control Features in Intel 64 Processor (R/W)	If any one enumeration condition for defined bit field holds.

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal	,	·	
		0	Lock bit (R/W0): (1 = locked). When set, locks this MSR from being written; writes to this bit will result in GP(0).	If any one enumeration condition for defined bit field position greater than bit 0 holds.
			Note: Once the Lock bit is set, the contents of this register cannot be modified. Therefore the lock bit must be set after configuring support for Intel Virtualization Technology and prior to transferring control to an option ROM or the OS. Hence, once the Lock bit is set, the entire IA32_FEATURE_CONTROL contents are preserved across RESET when PWRGOOD is not deasserted.	
		1	Enable VMX inside SMX operation (R/WL) This bit enables a system executive to use VMX in conjunction with SMX to support Intel® Trusted Execution Technology.	If CPUID.01H:ECX[5] = 1 && CPUID.01H:ECX[6] = 1
			BIOS must set this bit only when the CPUID function 1 returns VMX feature flag and SMX feature flag set (ECX bits 5 and 6 respectively).	
		2	Enable VMX outside SMX operation (R/WL) This bit enables VMX for a system executive that does not require SMX. BIOS must set this bit only when the CPUID function 1 returns the VMX feature flag set (ECX bit 5).	If CPUID.01H:ECX[5] = 1
		7:3	Reserved	
		14:8	SENTER Local Function Enables (R/WL) When set, each bit in the field represents an enable control for a corresponding SENTER function. This field is supported only if CPUID.1:ECX.[bit 6] is set.	If CPUID.01H:ECX[6] = 1
		15	SENTER Global Enable (R/WL) This bit must be set to enable SENTER leaf functions. This bit is supported only if CPUID.1:ECX.[bit 6] is set.	If CPUID.01H:ECX[6] = 1
		16	Reserved	
		17	SGX Launch Control Enable (R/WL)	If CPUID.(EAX=07H,
			This bit must be set to enable runtime reconfiguration of SGX Launch Control via the IA32_SGXLEPUBKEYHASHn MSR.	ECX=0H): ECX[30] = 1
		18	SGX Global Enable (R/WL) This bit must be set to enable SGX leaf functions.	If CPUID.(EAX=07H, ECX=0H): EBX[2] = 1
		19	Reserved	

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment	
Hex	Decimal				
		20	LMCE On (R/WL) When set, system software can program the MSRs associated with LMCE to configure delivery of some machine check exceptions to a single logical processor.	If IA32_MCG_CAP[27] = 1	
		63:21	Reserved		
3BH	59	IA32_TSC_ADJUST	Per Logical Processor TSC Adjust (R/Write to clear)	If CPUID.(EAX=07H, ECX=0H): EBX[1] = 1	
		63:0	THREAD_ADJUST Local offset value of the IA32_TSC for a logical processor. Reset value is zero. A write to IA32_TSC will modify the local offset in IA32_TSC_ADJUST and the content of IA32_TSC, but does not affect the internal invariant TSC hardware.		
48H	72	IA32_SPEC_CTRL	Speculation Control (R/W) The MSR bits are defined as logical processor scope. On some core implementations, the bits may impact sibling logical processors on the same core. This MSR has a value of 0 after reset and is unaffected by INIT# or SIPI#.	If any one of the enumeration conditions for defined bit field positions holds.	
		0	Indirect Branch Restricted Speculation (IBRS). Restricts speculation of indirect branch.	If CPUID.(EAX=07H, ECX=0):EDX[26]=1	
		1	Single Thread Indirect Branch Predictors (STIBP). Prevents indirect branch predictions on all logical processors on the core from being controlled by any sibling logical processor in the same core.	If CPUID.(EAX=07H, ECX=0):EDX[27]=1	
			2	Speculative Store Bypass Disable (SSBD) delays speculative execution of a load until the addresses for all older stores are known.	If CPUID.(EAX=07H, ECX=0):EDX[31]=1
		3	IPRED_DIS_U If 1, enables IPRED_DIS control for CPL3.	If CPUID.(EAX=07H, ECX=2):EDX[1]=1	
		4	IPRED_DIS_S If 1, enables IPRED_DIS control for CPLO/1/2.	If CPUID.(EAX=07H, ECX=2):EDX[1]=1	
		5	RRSBA_DIS_U If 1, disables RRSBA behavior for CPL3.	If CPUID.(EAX=07H, ECX=2):EDX[2]=1	
		6	RRSBA_DIS_S If 1, disables RRSBA behavior for CPL0/1/2.	If CPUID.(EAX=07H, ECX=2):EDX[2]=1	

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		7	PSFD If 1, disables Fast Store Forwarding Predictor. Note that setting bit 2 (SSBD) also disables this.	If CPUID.(EAX=07H, ECX=2):EDX[0]=1
		8	DDPD_U If 1, disables the Data Dependent Prefetcher that examines data values in memory while CPL = 3. Note that setting bit 2 (SSBD) also disables this.	If CPUID.(EAX=07H, ECX=2):EDX[3]=1
		9	Reserved	
		10	BHI_DIS_S When '1, enables BHI_DIS_S behavior.	If CPUID.(EAX=07H, ECX=2):EDX[4]=1
		63:11	Reserved	
49H	73	IA32_PRED_CMD	Prediction Command (WO) Gives software a way to issue commands that affect the state of predictors.	If any one of the enumeration conditions for defined bit field positions holds.
		0	Indirect Branch Prediction Barrier (IBPB)	If CPUID.(EAX=07H, ECX=0):EDX[26]=1
		63:1	Reserved	
4EH	78	IA32_PPIN_CTL	Protected Processor Inventory Number Enable Control (R/W)	If CPUID.(EAX=07H, ECX=01H):EBX[0]=1 ¹
		0	LockOut (R/W0)	
			If 0, indicates that further writes to IA32_PPIN_CTL is allowed.	
			If 1, indicates that further writes to IA32_PPIN_CTL is disallowed. Writing 1 to this bit is only permitted if the Enable_PPIN bit is clear.	
			The Privileged System Software Inventory Agent should read IA32_PPIN_CTL[bit 1] to determine if IA32_PPIN is accessible.	
			The Privileged System Software Inventory Agent is not expected to write to this MSR.	
		1	Enable_PPIN (R/W)	
			If 1, indicates that IA32_PPIN is accessible using RDMSR.	
			If O, indicates that IA32_PPIN is inaccessible using RDMSR. Any attempt to read IA32_PPIN will cause #GP.	
		63:2	Reserved	

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
4FH	79	IA32_PPIN	Protected Processor Inventory Number (R/O)	If CPUID.(EAX=07H, ECX=01H):EBX[0]=1 ¹
		63:0	Protected Processor Inventory Number (R/O)	
			A unique value within a given CPUID family/model/stepping signature that a privileged inventory initialization agent can access to identify each physical processor, when access to IA32_PPIN is enabled. Access to IA32_PPIN is permitted only if IA32_PPIN_CTL[bits 1:0] = '10b'.	
79H	121	IA32_BIOS_UPDT_TRIG	BIOS Update Trigger (W)	06_01H
		(BIOS_UPDT_TRIG)	Executing a WRMSR instruction to this MSR causes a microcode update to be loaded into the processor. See Section 10.11.6, "Microcode Update Loader."	
			A processor may prevent writing to this MSR when loading guest states on VM entries or saving guest states on VM exits.	
8BH	139	IA32_BIOS_SIGN_ID	BIOS Update Signature (R/W)	06_01H
		(BIOS_SIGN/BBL_CR_D3)	Returns the microcode update signature following the execution of CPUID.01H.	
			A processor may prevent writing to this MSR when loading guest states on VM entries or saving guest states on VM exits.	
		31:0	Reserved	
		63:32	It is recommended that this field be pre- loaded with zero prior to executing CPUID.	
			If the field remains zero following the execution of CPUID, this indicates that no microcode update is loaded. Any non-zero value is the microcode update signature.	

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
8CH	140	IA32_SGXLEPUBKEYHASH0	IA32_SGXLEPUBKEYHASH[63:0] (R/W) Bits 63:0 of the SHA256 digest of the SIGSTRUCT.MODULUS for SGX Launch Enclave. On reset, the default value is the digest of Intel's signing key.	
8DH	141	IA32_SGXLEPUBKEYHASH1	IA32_SGXLEPUBKEYHASH[127:64] (R/W) Bits 127:64 of the SHA256 digest of the SIGSTRUCT.MODULUS for SGX Launch Enclave. On reset, the default value is the digest of Intel's signing key.	Read permitted If CPUID.(EAX=12H,ECX=0H): EAX[0]=1 && CPUID.(EAX=07H, ECX=0H):ECX[30]=1. Write permitted if
8EH	142	IA32_SGXLEPUBKEYHASH2	IA32_SGXLEPUBKEYHASH[191:128] (R/W) Bits 191:128 of the SHA256 digest of the SIGSTRUCT.MODULUS for SGX Launch Enclave. On reset, the default value is the digest of Intel's signing key.	CPUID.(EAX=12H,ECX=0H): EAX[0]=1 && IA32_FEATURE_CONTROL[17] = 1 && IA32_FEATURE_CONTROL[0] = 1.
8FH	143	IA32_SGXLEPUBKEYHASH3	IA32_SGXLEPUBKEYHASH[255:192] (R/W) Bits 255:192 of the SHA256 digest of the SIGSTRUCT.MODULUS for SGX Launch Enclave. On reset, the default value is the digest of Intel's signing key.	0] - 1.
9BH	155	IA32_SMM_MONITOR_CTL	SMM Monitor Configuration (R/W)	If CPUID.01H: ECX[5]=1 CPUID.01H: ECX[6] = 1
		0	Valid (R/W)	
		1	Reserved	
		2	Controls SMI unblocking by VMXOFF (see Section 32.14.4).	If IA32_VMX_MISC[28]
		11:3	Reserved	
		31:12	MSEG Base (R/W)	
		63:32	Reserved	
9EH	158	IA32_SMBASE	Base address of the logical processor's SMRAM image (R/O, SMM only).	If IA32_VMX_MISC[15]
BCH	188	IA32_MISC_PACKAGE_CTLS	Power Filtering Control (R/W) This MSR has a value of 0 after reset and is unaffected by INIT# or SIPI#.	If IA32_ARCH_CAPABILITIES [10] = 1
		0	ENERGY_FILTERING_ENABLE (R/W)	If
			If set, RAPL MSRs report filtered processor power consumption data.	IA32_ARCH_CAPABILITIES [11] = 1
			This bit can be changed from 0 to 1, but cannot be changed from 1 to 0. After setting, all attempts to clear it are ignored until the next processor reset.	
		63:1	Reserved.	

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
BDH	189	IA32_XAPIC_DISABLE_STATUS	xAPIC Disable Status (R/O)	If CPUID.(EAX-07H, ECX=0):EDX[29]=1 and IA32_ARCH_CAPABILITIES [21] = 1
		0	LEGACY_XAPIC_DISABLED	
			When set, indicates that the local APIC is in x2APIC mode (IA32_APIC_BASE.EXTD = 1) and that attempts to clear IA32_APIC_BASE.EXTD will fail (e.g., WRMSR will #GP).	
		63:1	Reserved	
C1H	193	IA32_PMC0 (PERFCTR0)	General Performance Counter 0 (R/W)	If CPUID.OAH: EAX[15:8] > 0
C2H	194	IA32_PMC1 (PERFCTR1)	General Performance Counter 1 (R/W)	If CPUID.OAH: EAX[15:8] > 1
СЗН	195	IA32_PMC2	General Performance Counter 2 (R/W)	If CPUID.OAH: EAX[15:8] > 2
C4H	196	IA32_PMC3	General Performance Counter 3 (R/W)	If CPUID.OAH: EAX[15:8] > 3
C5H	197	IA32_PMC4	General Performance Counter 4 (R/W)	If CPUID.OAH: EAX[15:8] > 4
С6Н	198	IA32_PMC5	General Performance Counter 5 (R/W)	If CPUID.OAH: EAX[15:8] > 5
C7H	199	IA32_PMC6	General Performance Counter 6 (R/W)	If CPUID.OAH: EAX[15:8] > 6
C8H	200	IA32_PMC7	General Performance Counter 7 (R/W)	If CPUID.OAH: EAX[15:8] > 7
CFH	207	IA32_CORE_CAPABILITIES	IA32 Core Capabilities Register	If CPUID.(EAX=07H, ECX=0):EDX[30] = 1
		63:0	Reserved.	No architecturally defined bits.
E1H	225	IA32_UMWAIT_CONTROL	UMWAIT Control (R/W)	
		0	CO.2 is not allowed by the OS. Value of "1" means all CO.2 requests revert to CO.1.	
		1	Reserved	
		31:2	Determines the maximum time in TSC-quanta that the processor can reside in either CO.1 or CO.2. A zero value indicates no maximum time. The maximum time value is a 32-bit value where the upper 30 bits come from this field and the lower two bits are zero.	
E7H	231	IA32_MPERF	TSC Frequency Clock Counter (R/Write to clear)	If CPUID.06H: ECX[0] = 1

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		63:0	CO_MCNT: CO TSC Frequency Clock Count Increments at fixed interval (relative to TSC freq.) when the logical processor is in CO. Cleared upon overflow / wrap-around of IA32_APERF.	
E8H	232	IA32_APERF	Actual Performance Clock Counter (R/Write to clear)	If CPUID.06H: ECX[0] = 1
		63:0	CO_ACNT: CO Actual Frequency Clock Count Accumulates core clock counts at the coordinated clock frequency, when the logical processor is in CO. Cleared upon overflow / wrap-around of IA32_MPERF.	
FEH	254	IA32_MTRRCAP (MTRRcap)	MTRR Capability (R/O) See Section 12.11.2.1, "IA32_MTRR_DEF_TYPE MSR."	06_01H
		7:0	VCNT: The number of variable memory type ranges in the processor.	
		8	Fixed range MTRRs are supported when set.	
		9	Reserved	
		10	WC Supported when set.	
		11	SMRR Supported when set.	
		12	PRMRR supported when set.	
		63:13	Reserved	
10AH	266	IA32_ARCH_CAPABILITIES	Enumeration of Architectural Features (R/O)	If CPUID.(EAX=07H, ECX=0):EDX[29]=1
		0	RDCL_NO: The processor is not susceptible to Rogue Data Cache Load (RDCL).	
		1	IBRS_ALL: The processor supports enhanced IBRS.	
		2	RSBA: The processor supports RSB Alternate. Alternative branch predictors may be used by RET instructions when the RSB is empty. SW using retpoline may be affected by this behavior.	
		3	SKIP_L1DFL_VMENTRY: A value of 1 indicates the hypervisor need not flush the L1D on VM entry.	
		4	SSB_NO: Processor is not susceptible to Speculative Store Bypass.	
		5	MDS_NO: Processor is not susceptible to Microarchitectural Data Sampling (MDS).	

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		6	IF_PSCHANGE_MC_NO: The processor is not susceptible to a machine check error due to modifying the size of a code page without TLB invalidation.	
		7	TSX_CTRL: If 1, indicates presence of IA32_TSX_CTRL MSR.	
		8	TAA_NO: If 1, processor is not affected by TAA.	
		9	Reserved	
		10	MISC_PACKAGE_CTLS: The processor supports IA32_MISC_PACKAGE_CTLS MSR.	
		11	ENERGY_FILTERING_CTL: The processor supports setting and reading the IA32_MISC_PACKAGE_CTLS[0] (ENERGY_FILTERING_ENABLE) bit.	
		12	DOITM: If 1, the processor supports Data Operand Independent Timing Mode.	
		13	SBDR_SSDP_NO: The processor is not affected by either the Shared Buffers Data Read (SBDR) vulnerability or the Sideband Stale Data Propagator (SSDP).	
		14	FBSDP_NO: The processor is not affected by the Fill Buffer Stale Data Propagator (FBSDP).	
		15	PSDP_NO: The processor is not affected by vulnerabilities involving the Primary Stale Data Propagator (PSDP).	
		16	Reserved	
		17	FB_CLEAR: If 1, the processor supports overwrite of fill buffer values as part of MD_CLEAR operations with the VERW instruction.	
		18	FB_CLEAR_CTRL: If 1, the processor supports the IA32_MCU_OPT_CTRL MSR and allows software to set bit 3 of that MSR (FB_CLEAR_DIS).	
		19	RRSBA: A value of 1 indicates the processor may have the RRSBA alternate prediction behavior, if not disabled by RRSBA_DIS_U or RRSBA_DIS_S.	
		20	BHI_NO: A value of 1 indicates BHI_NO branch prediction behavior, regardless of the value of IA32_SPEC_CTRL[BHI_DIS_S] MSR bit.	

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		21	XAPIC_DISABLE_STATUS: Enumerates that the IA32_XAPIC_DISABLE_STATUS MSR exists, and that bit 0 specifies whether the legacy xAPIC is disabled and APIC state is locked to x2APIC.	
		22	Reserved	
		23	OVERCLOCKING_STATUS: If set, the IA32_OVERCLOCKING_STATUS MSR exists.	
		24	PBRSB_NO: If 1, the processor is not affected by issues related to Post-Barrier Return Stack Buffer Predictions.	
		63:25	Reserved	
10BH	267	IA32_FLUSH_CMD	Flush Command (WO) Gives software a way to invalidate structures with finer granularity than other architectural methods.	If any one of the enumeration conditions for defined bit field positions holds.
		0	L1D_FLUSH: Writeback and invalidate the L1 data cache.	If CPUID.(EAX=07H, ECX=0):EDX[28]=1
		63:1	Reserved	
10FH	271	IA32_TSX_FORCE_ABORT	TSX Force Abort	If CPUID.(EAX=07H, ECX=0):EDX[13]=1
		0	RTM_FORCE_ABORT	R/W, Default: 0
			If 1, all RTM transactions abort with EAX code 0.	If CPUID.(EAX=07H,ECX=0): EDX[11]=1, bit 0 is always 1 and writes to change it are ignored.
				If SDV_ENABLE_RTM is 1, bit 0 is always 0 and writes to change it are ignored.
		1	TSX_CPUID_CLEAR	R/W, Default: 0
			When set, CPUID.(EAX=07H,ECX=0):EBX[11]=0 and CPUID.(EAX=07H,ECX=0):EBX[4]=0.	Can be set only if CPUID.(EAX=07H,ECX=0): EDX[11]=1 or if SDV_ENABLE_RTM is 1.
		2	SDV_ENABLE_RTM	R/W, Default: 0
			When set, CPUID.(EAX=07H,ECX=0):EDX[11]=0 and the processor may not force abort RTM. This unsupported mode should only be used for software development and not for production usage.	If 0, can be set only if CPUID.(EAX=07H,ECX=0): EDX[11]=1.
		63:3	Reserved	

Table 2-2. IA-32 Architectural MSRs (Contd.)

	jister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
122H	290	IA32_TSX_CTRL	IA32_TSX_CTRL	Thread scope. Not architecturally serializing. Available when CPUID.ARCH_CAP(EAX=7H, ECX = 0):EDX[29] = 1 and IA32_ARCH_CAPABILITIES. bit 7 = 1.
		0	RTM_DISABLE	
			When set to 1, XBEGIN will always abort with EAX code 0.	
		1	TSX_CPUID_CLEAR	
			When set to 1, CPUID.07H.EBX.RTM [bit 11] and CPUID.07H.EBX.HLE [bit 4] report 0.	
			When set to 0 and the SKU supports TSX, these bits will return 1.	
		63:2	Reserved	
123H	291	IA32_MCU_OPT_CTRL	Microcode Update Option Control (R/W)	If CPUID.(EAX=07H, ECX=0):EDX[9]=1 or IA32_ARCH_CAPABILITIES [18] = 1 or IA32_ARCH_CAPABILITIES. FB_CLEAR_CTRL=1
		0	RNGDS_MITG_DIS (R/W)	If CPUID.(EAX=07H,
	If 0 (default), SRBDS mitigat for RDRAND and RDSEED.	If O (default), SRBDS mitigation is enabled for RDRAND and RDSEED.	ECX=0):EDX[9]=1	
			If 1, SRBDS mitigation is disabled for RDRAND and RDSEED executed outside of Intel SGX enclaves.	
		1	RTM_ALLOW	Read/Write
			If 0, XBEGIN will always abort with EAX code 0.	Setting RTM_LOCKED prevents writes to this bit.
			If 1, XBEGIN behavior depends on the value of IA32_TSX_CTRL[RTM_DISABLE].	
		2	RTM_LOCKED	Read-Only status bit
			When 1, RTM_ALLOW is locked at zero, writes to RTM_ALLOW will be ignored.	
		3	FB_CLEAR_DIS	If
			If 1, prevents the VERW instruction from performing an FB_CLEAR action.	IA32_ARCH_CAPABILITIES. FB_CLEAR_CTRL=1
		63:4	Reserved	
174H	372	IA32_SYSENTER_CS	SYSENTER_CS_MSR (R/W)	06_01H
		15:0	CS Selector.	
		31:16	Not used.	Can be read and written.

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		63:32	Not used.	Writes ignored; reads return zero.
175H	373	IA32_SYSENTER_ESP	SYSENTER_ESP_MSR (R/W)	06_01H
176H	374	IA32_SYSENTER_EIP	SYSENTER_EIP_MSR (R/W)	06_01H
179H	377	IA32_MCG_CAP (MCG_CAP)	Global Machine Check Capability (R/O)	06_01H
		7:0	Count: Number of reporting banks.	
		8	MCG_CTL_P: IA32_MCG_CTL is present if this bit is set.	
		9	MCG_EXT_P: Extended machine check state registers are present if this bit is set.	
		10	MCP_CMCI_P: Support for corrected MC error event is present.	06_01H
		11	MCG_TES_P: Threshold-based error status register are present if this bit is set.	
		15:12	Reserved	
		23:16	MCG_EXT_CNT: Number of extended machine check state registers present.	
		24	MCG_SER_P: The processor supports software error recovery if this bit is set.	
		25	Reserved	
		26	MCG_ELOG_P: Indicates that the processor allows platform firmware to be invoked when an error is detected so that it may provide additional platform specific information in an ACPI format "Generic Error Data Entry" that augments the data included in machine check bank registers.	06_3EH
		27	MCG_LMCE_P: Indicates that the processor supports extended state in IA32_MCG_STATUS and associated MSR necessary to configure Local Machine Check Exception (LMCE).	06_3EH
		63:28	Reserved	
17AH	378	IA32_MCG_STATUS (MCG_STATUS)	Global Machine Check Status (R/W0)	06_01H
		(MCd_31A103)	RIPV. Restart IP valid.	06_01H
		1	EIPV. Error IP valid.	06_01H
		2	MCIP. Machine check in progress.	06_01H
		3	LMCE_S	If
		63:4	Reserved	

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
17BH	379	IA32_MCG_CTL (MCG_CTL)	Global Machine Check Control (R/W)	If IA32_MCG_CAP.CTL_P[8] =1
180H- 185H	384- 389	Reserved		06_0EH ²
186H	390	IA32_PERFEVTSEL0 (PERFEVTSEL0)	Performance Event Select Register 0 (R/W)	If CPUID.OAH: EAX[15:8] > 0
		7:0	Event Select: Selects a performance event logic unit.	
		15:8	UMask: Qualifies the microarchitectural condition to detect on the selected event logic.	
		16	USR: Counts while in privilege level is not ring 0.	
		17	OS: Counts while in privilege level is ring 0.	
		18	Edge: Enables edge detection if set.	
		19	PC: Enables pin control.	
		20	INT: Enables interrupt on counter overflow.	
		21	AnyThread: When set to 1, it enables counting the associated event conditions occurring across all logical processors sharing a processor core. When set to 0, the counter only increments the associated event conditions occurring in the logical processor which programmed the MSR.	
		22	EN: Enables the corresponding performance counter to commence counting when this bit is set.	
		23	INV: Invert the CMASK.	
		31:24	CMASK: When CMASK is not zero, the corresponding performance counter increments each cycle if the event count is greater than or equal to the CMASK.	
		63:32	Reserved	
187H	391	IA32_PERFEVTSEL1 (PERFEVTSEL1)	Performance Event Select Register 1 (R/W)	If CPUID.OAH: EAX[15:8] > 1
188H	392	IA32_PERFEVTSEL2	Performance Event Select Register 2 (R/W)	If CPUID.OAH: EAX[15:8] > 2
189H	393	IA32_PERFEVTSEL3	Performance Event Select Register 3 (R/W)	If CPUID.OAH: EAX[15:8] > 3
18AH	394	IA32_PERFEVTSEL4	Performance Event Select Register 4 (R/W)	If CPUID.OAH: EAX[15:8] > 4
18BH	395	IA32_PERFEVTSEL5	Performance Event Select Register 5 (R/W)	If CPUID.OAH: EAX[15:8] > 5

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
18CH	396	IA32_PERFEVTSEL6	Performance Event Select Register 6 (R/W)	If CPUID.OAH: EAX[15:8] > 6
18DH	397	IA32_PERFEVTSEL7	Performance Event Select Register 7 (R/W)	If CPUID.OAH: EAX[15:8] > 7
18AH- 194H	394- 404	Reserved		06_0EH ³
195H	405	IA32_OVERCLOCKING_STATUS	Overclocking Status (R/O) IA32_ARCH_CAPABILITIES[bit 23] enumerates support for this MSR.	
		0	Overclocking Utilized	
			Indicates if specific forms of overclocking have been enabled on this boot or reset cycle: 0 indicates no, 1 indicates yes.	
		1	Undervolt Protection Indicates if the "Dynamic OC Undervolt Protection" security feature is active: 0 indicates disabled, 1indicates enabled.	
		2	Overclocking Secure Status Indicates that overclocking capabilities have been unlocked by BIOS, with or without overclocking: 0 indicates Not Secured, 1 indicates Secure.	
		63:4	Reserved	
196H- 197H	406- 407	Reserved		06_0EH ³
198H	408	IA32_PERF_STATUS	Current Performance Status (R/O) See Section 15.1.1, "Software Interface For Initiating Performance State Transitions".	0F_03H
		15:0	Current performance State Value.	
		63:16	Reserved	
199H	409	IA32_PERF_CTL	Performance Control MSR (R/W) Software makes a request for a new Performance state (P-State) by writing this MSR. See Section 15.1.1, "Software Interface For Initiating Performance State Transitions".	0F_03H
		15:0	Target performance State Value.	
		31:16	Reserved	
		32	Intel® Dynamic Acceleration Technology Engage (R/W) When set to 1: Disengages Intel Dynamic	06_0FH (Mobile only)
			Acceleration Technology.	
		63:33	Reserved	

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
19AH 410	IA32_CLOCK_MODULATION	Clock Modulation Control (R/W) See Section 15.8.3, "Software Controlled Clock Modulation."	If CPUID.01H:EDX[22] = 1	
		0	Extended On-Demand Clock Modulation Duty Cycle.	If CPUID.06H:EAX[5] = 1
		3:1	On-Demand Clock Modulation Duty Cycle: Specific encoded values for target duty cycle modulation.	If CPUID.01H:EDX[22] = 1
		4	On-Demand Clock Modulation Enable: Set 1 to enable modulation.	If CPUID.01H:EDX[22] = 1
		63:5	Reserved	
19BH	411	IA32_THERM_INTERRUPT	Thermal Interrupt Control (R/W)	If CPUID.01H:EDX[22] = 1
			Enables and disables the generation of an interrupt on temperature transitions detected with the processor's thermal sensors and thermal monitor.	
			See Section 15.8.2, "Thermal Monitor."	
		0	High-Temperature Interrupt Enable	If CPUID.01H:EDX[22] = 1
		1	Low-Temperature Interrupt Enable	If CPUID.01H:EDX[22] = 1
		2	PROCHOT# Interrupt Enable	If CPUID.01H:EDX[22] = 1
		3	FORCEPR# Interrupt Enable	If CPUID.01H:EDX[22] = 1
		4	Critical Temperature Interrupt Enable	If CPUID.01H:EDX[22] = 1
		7:5	Reserved	
		14:8	Threshold #1 Value	If CPUID.01H:EDX[22] = 1
		15	Threshold #1 Interrupt Enable	If CPUID.01H:EDX[22] = 1
		22:16	Threshold #2 Value	If CPUID.01H:EDX[22] = 1
		23	Threshold #2 Interrupt Enable	If CPUID.01H:EDX[22] = 1
		24	Power Limit Notification Enable	If CPUID.06H:EAX[4] = 1
		25	Hardware Feedback Notification Enable	If CPUID.06H:EAX[24] = 1
		63:26	Reserved	
19CH	412	IA32_THERM_STATUS	Thermal Status Information (R/O)	If CPUID.01H:EDX[22] = 1
			Contains status information about the processor's thermal sensor and automatic thermal monitoring facilities.	
			See Section 15.8.2, "Thermal Monitor".	
		0	Thermal Status (R/O)	If CPUID.01H:EDX[22] = 1
		1	Thermal Status Log (R/W)	If CPUID.01H:EDX[22] = 1
		2	PROCHOT # or FORCEPR# event (R/O)	If CPUID.01H:EDX[22] = 1
		3	PROCHOT # or FORCEPR# log (R/WC0)	If CPUID.01H:EDX[22] = 1
		4	Critical Temperature Status (R/O)	If CPUID.01H:EDX[22] = 1

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		5	Critical Temperature Status log (R/WC0)	If CPUID.01H:EDX[22] = 1
		6	Thermal Threshold #1 Status (R/O)	If CPUID.01H:ECX[8] = 1
		7	Thermal Threshold #1 log (R/WCO)	If CPUID.01H:ECX[8] = 1
		8	Thermal Threshold #2 Status (R/O)	If CPUID.01H:ECX[8] = 1
		9	Thermal Threshold #2 log (R/WC0)	If CPUID.01H:ECX[8] = 1
		10	Power Limitation Status (R/0)	If CPUID.06H:EAX[4] = 1
		11	Power Limitation log (R/WC0)	If CPUID.06H:EAX[4] = 1
		12	Current Limit Status (R/O)	If CPUID.06H:EAX[7] = 1
		13	Current Limit log (R/WCO)	If CPUID.06H:EAX[7] = 1
		14	Cross Domain Limit Status (R/O)	If CPUID.06H:EAX[7] = 1
		15	Cross Domain Limit log (R/WCO)	If CPUID.06H:EAX[7] = 1
		22:16	Digital Readout (R/O)	If CPUID.06H:EAX[0] = 1
		26:23	Reserved	
		30:27	Resolution in Degrees Celsius (R/O)	If CPUID.06H:EAX[0] = 1
		31	Reading Valid (R/O)	If CPUID.06H:EAX[0] = 1
		63:32	Reserved	
1A0H	416	IA32_MISC_ENABLE	Enable Misc. Processor Features (R/W)	
			Allows a variety of processor functions to be enabled and disabled.	
		0	Fast-Strings Enable	OF_OH
			When set, the fast-strings feature (for REP MOVS and REP STORS) is enabled (default). When clear, fast-strings are disabled.	
		2:1	Reserved	
		3	Automatic Thermal Control Circuit Enable (R/W) 1 = Setting this bit enables the thermal control circuit (TCC) portion of the Intel Thermal Monitor feature. This allows the processor to automatically reduce power consumption in response to TCC activation. 0 = Disabled. Note: In some products clearing this bit might be ignored in critical thermal conditions, and TM1, TM2, and adaptive thermal throttling will still be activated. The default value of this field varies with product . See respective tables where default value is listed.	OF_OH
		6:4	Reserved	

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		7	Performance Monitoring Available (R) 1 = Performance monitoring enabled. 0 = Performance monitoring disabled.	OF_OH
		10:8	Reserved	
		11	Branch Trace Storage Unavailable (R/O) 1 = Processor doesn't support branch trace storage (BTS). 0 = BTS is supported.	OF_OH
		12	Processor Event Based Sampling (PEBS) Unavailable (R/O) 1 = PEBS is not supported. 0 = PEBS is supported.	06_0FH
		15:13	Reserved	
		16	Enhanced Intel SpeedStep Technology Enable (R/W)	If CPUID.01H: ECX[7] =1
			 0= Enhanced Intel SpeedStep Technology disabled. 1 = Enhanced Intel SpeedStep Technology enabled. 	
		17	Reserved	
		18	ENABLE MONITOR FSM (R/W) When this bit is set to 0, the MONITOR feature flag is not set (CPUID.01H:ECX[bit 3] = 0). This indicates that MONITOR/MWAIT are not supported.	0F_03H
			Software attempts to execute MONITOR/MWAIT will cause #UD when this bit is 0.	
			When this bit is set to 1 (default), MONITOR/MWAIT are supported (CPUID.01H:ECX[bit 3] = 1).	
			If the SSE3 feature flag ECX[0] is not set (CPUID.01H:ECX[bit 0] = 0), the OS must not attempt to alter this bit. BIOS must leave it in the default state. Writing this bit when the SSE3 feature flag is set to 0 may generate a #GP exception.	
		21:19	Reserved	

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		22	Limit CPUID Maxval (R/W)	0F_03H
			When this bit is set to 1, CPUID.00H returns a maximum value in EAX[7:0] of 2.	
			BIOS should contain a setup question that allows users to specify when the installed OS does not support CPUID functions greater than 2.	
			Before setting this bit, BIOS must execute the CPUID.OH and examine the maximum value returned in EAX[7:0]. If the maximum value is greater than 2, this bit is supported.	
			Otherwise, this bit is not supported. Setting this bit when the maximum value is not greater than 2 may generate a #GP exception.	
			Setting this bit may cause unexpected behavior in software that depends on the availability of CPUID leaves greater than 2.	
		23	xTPR Message Disable (R/W) When set to 1, xTPR messages are disabled. xTPR messages are optional messages that allow the processor to inform the chipset of its priority.	If CPUID.01H:ECX[14] = 1
		33:24	Reserved	
		34	XD Bit Disable (R/W)	If
			When set to 1, the Execute Disable Bit feature (XD Bit) is disabled and the XD Bit extended feature flag will be clear (CPUID.80000001H: EDX[20]=0).	CPUID.80000001H:EDX[20] = 1
			When set to a 0 (default), the Execute Disable Bit feature (if available) allows the OS to enable PAE paging and take advantage of data only pages.	
			BIOS must not alter the contents of this bit location, if XD bit is not supported. Writing this bit to 1 when the XD Bit extended feature flag is set to 0 may generate a #GP exception.	
		63:35	Reserved	
1B0H	432	IA32_ENERGY_PERF_BIAS	Performance Energy Bias Hint (R/W)	If CPUID.6H:ECX[3] = 1
		3:0	Power Policy Preference:	
			O indicates preference to highest performance.	
			15 indicates preference to maximize energy saving.	
		63:4	Reserved	

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
1B1H	433	IA32_PACKAGE_THERM_STATUS	Package Thermal Status Information (R/O) Contains status information about the package's thermal sensor. See Section 15.9, "Package Level Thermal Management."	If CPUID.06H: EAX[6] = 1
		0	Pkg Thermal Status (R/O)	
		1	Pkg Thermal Status Log (R/W)	
		2	Pkg PROCHOT # event (R/O)	
		3	Pkg PROCHOT # log (R/WC0)	
		4	Pkg Critical Temperature Status (R/O)	
		5	Pkg Critical Temperature Status Log (R/WCO)	
		6	Pkg Thermal Threshold #1 Status (R/O)	
		7	Pkg Thermal Threshold #1 Log (R/WC0)	
		8	Pkg Thermal Threshold #2 Status (R/O)	
		9	Pkg Thermal Threshold #1 Log (R/WC0)	
		10	Pkg Power Limitation Status (R/0)	
		11	Pkg Power Limitation Log (R/WC0)	
		15:12	Reserved	
		22:16	Pkg Digital Readout (R/O)	
		25:23	Reserved	
		26	Hardware Feedback Interface Structure Change Status	If CPUID.06H:EAX.[19] = 1
		63:27	Reserved	
1B2H	434	IA32_PACKAGE_THERM_INTERRUPT	Pkg Thermal Interrupt Control (R/W) Enables and disables the generation of an interrupt on temperature transitions detected with the package's thermal sensor. See Section 15.9, "Package Level Thermal Management."	If CPUID.06H: EAX[6] = 1
		0	Pkg High-Temperature Interrupt Enable	
		1	Pkg Low-Temperature Interrupt Enable	
		2	Pkg PROCHOT# Interrupt Enable	
		3	Reserved	
		4	Pkg Overheat Interrupt Enable	
		7:5	Reserved	
		14:8	Pkg Threshold #1 Value	
		15	Pkg Threshold #1 Interrupt Enable	
	<u> </u>	22:16	Pkg Threshold #2 Value	

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		23	Pkg Threshold #2 Interrupt Enable	
		24	Pkg Power Limit Notification Enable	
		25	Hardware Feedback Interrupt Enable	If CPUID.06H:EAX.[19] = 1
		63:26	Reserved	
1C4H	452	IA32_XFD	Extended Feature Disable Control (R/W) Controls which XSAVE-enabled features are temporarily disabled.	If CPUID.(EAX=0DH,ECX=1): EAX[4] = 1
			See Section 13.14 of the Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 1.	
1C5H	453	IA32_XFD_ERR	Extended Feature Disable Error Code (R/W) Reports which XSAVE-enabled features caused a fault due to being disabled.	If CPUID.(EAX=0DH,ECX=1): EAX[4] = 1
			See Section 13.14 of the Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 1.	
1D9H	473	IA32_DEBUGCTL	Trace/Profile Resource Control (R/W)	06_0EH
		(MSR_DEBUGCTLA, MSR_DEBUGCTLB)		
		0	LBR: Setting this bit to 1 enables the processor to record a running trace of the most recent branches taken by the processor in the LBR stack.	06_01H
		1	BTF: Setting this bit to 1 enables the processor to treat EFLAGS.TF as single-step on branches instead of single-step on instructions.	06_01H
		5:2	Reserved	
		6	TR: Setting this bit to 1 enables branch trace messages to be sent.	06_0EH
		7	BTS: Setting this bit enables branch trace messages (BTMs) to be logged in a BTS buffer.	06_0EH
		8	BTINT: When clear, BTMs are logged in a BTS buffer in circular fashion. When this bit is set, an interrupt is generated by the BTS facility when the BTS buffer is full.	06_0EH
		9	1: BTS_OFF_OS: When set, BTS or BTM is skipped if CPL = 0.	06_0FH
		10	BTS_OFF_USR: When set, BTS or BTM is skipped if CPL > 0.	06_0FH
		11	FREEZE_LBRS_ON_PMI: When set, the LBR stack is frozen on a PMI request.	If CPUID.01H: ECX[15] = 1 && CPUID.0AH: EAX[7:0] > 1

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal	12		
		12	FREEZE_PERFMON_ON_PMI: When set, each ENABLE bit of the global counter control MSR are frozen (address 38FH) on a PMI request.	If CPUID.01H: ECX[15] = 1 && CPUID.0AH: EAX[7:0] > 1
		13	ENABLE_UNCORE_PMI: When set, enables the logical processor to receive and generate PMI on behalf of the uncore.	06_1AH
		14	FREEZE_WHILE_SMM: When set, freezes perfmon and trace messages while in SMM.	If IA32_PERF_CAPABILITIES[12] = 1
		15	RTM_DEBUG: When set, enables DR7 debug bit on XBEGIN.	If (CPUID.(EAX=07H, ECX=0):EBX[11] = 1)
		63:16	Reserved	
1DDH	477	IA32_LER_FROM_IP	Last Event Record Source IP Register (R/W)	
		63:0	FROM_IP	Reset Value: 0
			The source IP of the recorded branch or event, in canonical form.	
1DEH	478	IA32_LER_TO_IP	Last Event Record Destination IP Register (R/W)	
		63:0	TO_IP The destination IP of the recorded branch or event, in canonical form.	Reset Value: 0
1E0H	480	IA32_LER_INFO	Last Event Record Info Register (R/W)	
		55:0	Undefined, may be zero or non-zero. Writes of non- zero values do not fault, but reads may return a different value.	Reset Value: 0
		59:56	BR_TYPE The branch type recorded by this LBR. Encodings match those of IA32_LBR_x_INFO.	Reset Value: 0
		60	Undefined, may be zero or non-zero. Writes of non- zero values do not fault, but reads may return a different value.	Reset Value: 0
		61	TSX_ABORT This LBR record is a TSX abort. On processors that do not support Intel® TSX (CPUID.07H.EBX.HLE[bit 4]=0 and CPUID.07H.EBX.RTM[bit 11]=0), this bit is undefined.	Reset Value: 0
		62	IN_TSX This LBR record records a branch that retired during a TSX transaction. On processors that do not support Intel® TSX (CPUID.07H.EBX.HLE[bit 4]=0 and CPUID.07H.EBX.RTM[bit 11]=0), this bit is undefined.	Reset Value: 0

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		63	MISPRED The recorded branch taken/not-taken resolution (for conditional branches) or target (for any indirect branch, including RETs) was mispredicted.	Reset Value: 0
1F2H	498	IA32_SMRR_PHYSBASE	SMRR Base Address (Writeable only in SMM) Base address of SMM memory range.	If IA32_MTRRCAP.SMRR[11] = 1
		7:0	Type. Specifies memory type of the range.	
		11:8	Reserved	
		31:12	PhysBase SMRR physical Base Address.	
		63:32	Reserved	
1F3H	499	IA32_SMRR_PHYSMASK	SMRR Range Mask (Writeable only in SMM) Range Mask of SMM memory range.	If IA32_MTRRCAP[SMRR] = 1
		10:0	Reserved	
		11	Valid Enable range mask.	
		31:12	PhysMask SMRR address range mask.	
		63:32	Reserved	
1F8H	504	IA32_PLATFORM_DCA_CAP	DCA Capability (R)	If CPUID.01H: ECX[18] = 1
1F9H	505	IA32_CPU_DCA_CAP	If set, CPU supports Prefetch-Hint type.	If CPUID.01H: ECX[18] = 1
1FAH	506	IA32_DCA_0_CAP	DCA type 0 Status and Control register.	If CPUID.01H: ECX[18] = 1
		0	DCA_ACTIVE: Set by HW when DCA is fuse- enabled and no defeatures are set.	
		2:1	TRANSACTION	
		6:3	DCA_TYPE	
		10:7	DCA_QUEUE_SIZE	
		12:11	Reserved	
		16:13	DCA_DELAY: Writes will update the register but have no HW side-effect.	
		23:17	Reserved	
		24	SW_BLOCK: SW can request DCA block by setting this bit.	
		25	Reserved	
		26	HW_BLOCK: Set when DCA is blocked by HW (e.g., CR0.CD = 1).	
		31:27	Reserved	

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
200H	512	IA32_MTRR_PHYSBASE0 (MTRRphysBase0)	See Section 12.11.2.3, "Variable Range MTRRs."	If IA32_MTRRCAP[7:0] > 0
201H	513	IA32_MTRR_PHYSMASKO	MTRRphysMask0	If IA32_MTRRCAP[7:0] > 0
202H	514	IA32_MTRR_PHYSBASE1	MTRRphysBase1	If IA32_MTRRCAP[7:0] > 1
203H	515	IA32_MTRR_PHYSMASK1	MTRRphysMask1	If IA32_MTRRCAP[7:0] > 1
204H	516	IA32_MTRR_PHYSBASE2	MTRRphysBase2	If IA32_MTRRCAP[7:0] > 2
205H	517	IA32_MTRR_PHYSMASK2	MTRRphysMask2	If IA32_MTRRCAP[7:0] > 2
206H	518	IA32_MTRR_PHYSBASE3	MTRRphysBase3	If IA32_MTRRCAP[7:0] > 3
207H	519	IA32_MTRR_PHYSMASK3	MTRRphysMask3	If IA32_MTRRCAP[7:0] > 3
208H	520	IA32_MTRR_PHYSBASE4	MTRRphysBase4	If IA32_MTRRCAP[7:0] > 4
209H	521	IA32_MTRR_PHYSMASK4	MTRRphysMask4	If IA32_MTRRCAP[7:0] > 4
20AH	522	IA32_MTRR_PHYSBASE5	MTRRphysBase5	If IA32_MTRRCAP[7:0] > 5
20BH	523	IA32_MTRR_PHYSMASK5	MTRRphysMask5	If IA32_MTRRCAP[7:0] > 5
20CH	524	IA32_MTRR_PHYSBASE6	MTRRphysBase6	If IA32_MTRRCAP[7:0] > 6
20DH	525	IA32_MTRR_PHYSMASK6	MTRRphysMask6	If IA32_MTRRCAP[7:0] > 6
20EH	526	IA32_MTRR_PHYSBASE7	MTRRphysBase7	If IA32_MTRRCAP[7:0] > 7
20FH	527	IA32_MTRR_PHYSMASK7	MTRRphysMask7	If IA32_MTRRCAP[7:0] > 7
210H	528	IA32_MTRR_PHYSBASE8	MTRRphysBase8	If IA32_MTRRCAP[7:0] > 8
211H	529	IA32_MTRR_PHYSMASK8	MTRRphysMask8	If IA32_MTRRCAP[7:0] > 8
212H	530	IA32_MTRR_PHYSBASE9	MTRRphysBase9	If IA32_MTRRCAP[7:0] > 9
213H	531	IA32_MTRR_PHYSMASK9	MTRRphysMask9	If IA32_MTRRCAP[7:0] > 9
250H	592	IA32_MTRR_FIX64K_00000	MTRRfix64K_00000	If CPUID.01H: EDX.MTRR[12] =1
258H	600	IA32_MTRR_FIX16K_80000	MTRRfix16K_80000	If CPUID.01H: EDX.MTRR[12] =1
259H	601	IA32_MTRR_FIX16K_A0000	MTRRfix16K_A0000	If CPUID.01H: EDX.MTRR[12] =1
268H	616	IA32_MTRR_FIX4K_C0000 (MTRRfix4K_C0000)	See Section 12.11.2.2, "Fixed Range MTRRs."	If CPUID.01H: EDX.MTRR[12] =1
269H	617	IA32_MTRR_FIX4K_C8000	MTRRfix4K_C8000	If CPUID.01H: EDX.MTRR[12] =1
26AH	618	IA32_MTRR_FIX4K_D0000	MTRRfix4K_D0000	If CPUID.01H: EDX.MTRR[12] =1
26BH	619	IA32_MTRR_FIX4K_D8000	MTRRfix4K_D8000	If CPUID.01H: EDX.MTRR[12] =1
26CH	620	IA32_MTRR_FIX4K_E0000	MTRRfix4K_E0000	If CPUID.01H: EDX.MTRR[12] =1
26DH	621	IA32_MTRR_FIX4K_E8000	MTRRfix4K_E8000	If CPUID.01H: EDX.MTRR[12] =1

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
26EH	622	IA32_MTRR_FIX4K_F0000	MTRRfix4K_F0000	If CPUID.01H: EDX.MTRR[12] =1
26FH	623	IA32_MTRR_FIX4K_F8000	MTRRfix4K_F8000	If CPUID.01H: EDX.MTRR[12] =1
277H	631	IA32_PAT	IA32_PAT (R/W)	If CPUID.01H: EDX.MTRR[16] =1
		2:0	PA0	
		7:3	Reserved	
		10:8	PA1	
		15:11	Reserved	
		18:16	PA2	
		23:19	Reserved	
		26:24	PA3	
		31:27	Reserved	
		34:32	PA4	
		39:35	Reserved	
		42:40	PA5	
		47:43	Reserved	
		50:48	PA6	
		55:51	Reserved	
		58:56	PA7	
		63:59	Reserved	
280H	640	IA32_MCO_CTL2	MSR to enable/disable CMCI capability for bank 0. (R/W)	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] >
			See Section 16.3.2.5, "IA32_MCi_CTL2 MSRs".	0
		14:0	Corrected error count threshold.	
		29:15	Reserved	
		30	CMCI_EN	
		63:31	Reserved	
281H	641	IA32_MC1_CTL2	(R/W) Same fields as IA32_MCO_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 1
282H	642	IA32_MC2_CTL2	(R/W) Same fields as IA32_MC0_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 2
283H	643	IA32_MC3_CTL2	(R/W) Same fields as IA32_MC0_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 3

Table 2-2. IA-32 Architectural MSRs (Contd.)

	jister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
284H	644	IA32_MC4_CTL2	(R/W) Same fields as IA32_MC0_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 4
285H	645	IA32_MC5_CTL2	(R/W) Same fields as IA32_MCO_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 5
286H	646	IA32_MC6_CTL2	(R/W) Same fields as IA32_MCO_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 6
287H	647	IA32_MC7_CTL2	(R/W) Same fields as IA32_MCO_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 7
288H	648	IA32_MC8_CTL2	(R/W) Same fields as IA32_MCO_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 8
289H	649	IA32_MC9_CTL2	(R/W) Same fields as IA32_MCO_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 9
28AH	650	IA32_MC10_CTL2	(R/W) Same fields as IA32_MCO_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 10
28BH	651	IA32_MC11_CTL2	(R/W) Same fields as IA32_MC0_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 11
28CH	652	IA32_MC12_CTL2	(R/W) Same fields as IA32_MCO_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 12
28DH	653	IA32_MC13_CTL2	(R/W) Same fields as IA32_MCO_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 13
28EH	654	IA32_MC14_CTL2	(R/W) Same fields as IA32_MCO_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 14
28FH	655	IA32_MC15_CTL2	(R/W) Same fields as IA32_MCO_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 15
290H	656	IA32_MC16_CTL2	(R/W) Same fields as IA32_MC0_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 16
291H	657	IA32_MC17_CTL2	(R/W) Same fields as IA32_MCO_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 17
292H	658	IA32_MC18_CTL2	(R/W) Same fields as IA32_MC0_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 18

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
293H	659	IA32_MC19_CTL2	(R/W) Same fields as IA32_MC0_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 19
294H	660	IA32_MC20_CTL2	(R/W) Same fields as IA32_MC0_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 20
295H	661	IA32_MC21_CTL2	(R/W) Same fields as IA32_MC0_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 21
296H	662	IA32_MC22_CTL2	(R/W) Same fields as IA32_MC0_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 22
297H	663	IA32_MC23_CTL2	(R/W) Same fields as IA32_MC0_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 23
298H	664	IA32_MC24_CTL2	(R/W) Same fields as IA32_MC0_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 24
299H	665	IA32_MC25_CTL2	(R/W) Same fields as IA32_MC0_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 25
29AH	666	IA32_MC26_CTL2	(R/W) Same fields as IA32_MC0_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 26
29BH	667	IA32_MC27_CTL2	(R/W) Same fields as IA32_MC0_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 27
29CH	668	IA32_MC28_CTL2	(R/W) Same fields as IA32_MC0_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 28
29DH	669	IA32_MC29_CTL2	(R/W) Same fields as IA32_MC0_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 29
29EH	670	IA32_MC30_CTL2	(R/W) Same fields as IA32_MC0_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 30
29FH	671	IA32_MC31_CTL2	(R/W) Same fields as IA32_MC0_CTL2.	If IA32_MCG_CAP[10] = 1 && IA32_MCG_CAP[7:0] > 31
2FFH	767	IA32_MTRR_DEF_TYPE	MTRRdefType (R/W)	If CPUID.01H: EDX.MTRR[12] =1
		2:0	Default Memory Type	
		9:3	Reserved	
		10	Fixed Range MTRR Enable	
		11	MTRR Enable	

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name) MSR/Bit Description	Comment	
Hex	Decimal			
		63:12	Reserved	
309H	777	IA32_FIXED_CTR0	Fixed-Function Performance Counter 0 (R/W): Counts Instr_Retired.Any.	If CPUID.OAH: EDX[4:0] > 0
30AH	778	IA32_FIXED_CTR1	Fixed-Function Performance Counter 1 (R/W): Counts CPU_CLK_Unhalted.Core.	If CPUID.OAH: EDX[4:0] > 1
30BH	779	IA32_FIXED_CTR2	Fixed-Function Performance Counter 2 (R/W): Counts CPU_CLK_Unhalted.Ref.	If CPUID.OAH: EDX[4:0] > 2
345H	837	IA32_PERF_CAPABILITIES	Read Only MSR that enumerates the existence of performance monitoring features. (R/O)	If CPUID.01H: ECX[15] = 1
		5:0	LBR format	
		6	PEBS Trap	
		7	PEBSSaveArchRegs	
		11:8	PEBS Record Format	
		12	1: Freeze while SMM is supported.	
		13	1: Full width of counter writable via IA32_A_PMCx.	
		14	PEBS_BASELINE	
		15	1: Performance metrics available.	
		16	1: PEBS output will be written into the Intel PT trace stream.	If CPUID.0x7.0.EBX[25]=1
		63:17	Reserved	
38DH	909	IA32_FIXED_CTR_CTRL	Fixed-Function Performance Counter Control (R/W)	If CPUID.OAH: EAX[7:0] > 1
			Counter increments while the results of ANDing respective enable bit in IA32_PERF_GLOBAL_CTRL with the corresponding OS or USR bits in this MSR is true.	
		0	ENO_OS: Enable Fixed Counter 0 to count while CPL = 0.	
		1	ENO_Usr: Enable Fixed Counter 0 to count while CPL > 0.	
		2	AnyThrO: When set to 1, it enables counting the associated event conditions occurring across all logical processors sharing a processor core. When set to 0, the counter only increments the associated event conditions occurring in the logical processor which programmed the MSR.	If CPUID.OAH:EAX[7:0] > 2 && CPUID.OAH:EDX[15]=0
		3	ENO_PMI: Enable PMI when fixed counter 0 overflows.	
		4	EN1_OS: Enable Fixed Counter 1 to count while CPL = 0.	

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		5	EN1_Usr: Enable Fixed Counter 1to count while CPL > 0.	
		6	AnyThr1: When set to 1, it enables counting the associated event conditions occurring across all logical processors sharing a processor core. When set to 0, the counter only increments the associated event conditions occurring in the logical processor which programmed the MSR.	If CPUID.OAH:EAX[7:0] > 2 && CPUID.OAH:EDX[15]=0
		7	EN1_PMI: Enable PMI when fixed counter 1 overflows.	
		8	EN2_OS: Enable Fixed Counter 2 to count while CPL = 0.	
		9	EN2_Usr: Enable Fixed Counter 2 to count while CPL > 0.	
		10	AnyThr2: When set to 1, it enables counting the associated event conditions occurring across all logical processors sharing a processor core. When set to 0, the counter only increments the associated event conditions occurring in the logical processor which programmed the MSR.	If CPUID.OAH:EAX[7:0] > 2 && CPUID.OAH:EDX[15]=0
		11	EN2_PMI: Enable PMI when fixed counter 2 overflows.	
		12	EN3_OS: Enable Fixed Counter 3 to count while CPL = 0.	
		13	EN3_Usr: Enable Fixed Counter 3 to count while CPL > 0.	
		14	Reserved	
		15	EN3_PMI: Enable PMI when fixed counter 3 overflows.	
		63:16	Reserved	
38EH	910	IA32_PERF_GLOBAL_STATUS	Global Performance Counter Status (R/O)	If CPUID.0AH: EAX[7:0] > 0 II (CPUID.(EAX=07H, ECX=0):EBX[25] = 1 && CPUID.(EAX=014H, ECX=0):ECX[0] = 1)
		0	Ovf_PMCO: Overflow status of IA32_PMCO.	If CPUID.OAH: EAX[15:8] > 0
		1	Ovf_PMC1: Overflow status of IA32_PMC1.	If CPUID.OAH: EAX[15:8] > 1
		2	Ovf_PMC2: Overflow status of IA32_PMC2.	If CPUID.OAH: EAX[15:8] > 2
		3	Ovf_PMC3: Overflow status of IA32_PMC3.	If CPUID.OAH: EAX[15:8] > 3

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		n	Ovf_PMCn: Overflow status of IA32_PMCn.	If CPUID.OAH: EAX[15:8] > n
		31:n+1	Reserved	
		32	Ovf_FixedCtr0: Overflow status of IA32_FIXED_CTR0.	If CPUID.OAH: EAX[7:0] > 1
		33	Ovf_FixedCtr1: Overflow status of IA32_FIXED_CTR1.	If CPUID.OAH: EAX[7:0] > 1
		34	Ovf_FixedCtr2: Overflow status of IA32_FIXED_CTR2.	If CPUID.OAH: EAX[7:0] > 1
		47:35	Reserved	
		48	OVF_PERF_METRICS: If this bit is set, it indicates that PERF_METRIC counter has overflowed and a PMI is triggered; however, an overflow of fixed counter 3 should normally happen first. If this bit is clear no overflow occurred.	
		54:49	Reserved	
		55	Trace_ToPA_PMI: A PMI occurred due to a ToPA entry memory buffer that was completely filled.	If CPUID.(EAX=07H, ECX=0):EBX[25] = 1 && CPUID.(EAX=014H, ECX=0):ECX[0] = 1
		57:56	Reserved	
		58	LBR_Frz. LBRs are frozen due to: IA32_DEBUGCTL.FREEZE_LBR_ON_PMI=1. The LBR stack overflowed.	If CPUID.0AH: EAX[7:0] > 3
		59	CTR_Frz. Performance counters in the core PMU are frozen due to: IA32_DEBUGCTL.FREEZE_PERFMON_ON_PMI=1. One or more core PMU counters overflowed.	If CPUID.OAH: EAX[7:0] > 3
		60	ASCI: Data in the performance counters in the core PMU may include contributions from the direct or indirect operation Intel SGX to protect an enclave.	If CPUID.(EAX=07H, ECX=0):EBX[2] = 1
		61	Ovf_Uncore: Uncore counter overflow status.	If CPUID.OAH: EAX[7:0] > 2
		62	OvfBuf: DS SAVE area Buffer overflow status.	If CPUID.OAH: EAX[7:0] > 0
		63	CondChgd: Status bits of this register have changed.	If CPUID.OAH: EAX[7:0] > 0

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
38FH	911	IA32_PERF_GLOBAL_CTRL	Global Performance Counter Control (R/W) Counter increments while the result of ANDing the respective enable bit in this MSR with the corresponding OS or USR bits in the general-purpose or fixed counter control MSR is true.	If CPUID.0AH: EAX[7:0] > 0
		0	EN_PMCO	If CPUID.OAH: EAX[15:8] > 0
		1	EN_PMC1	If CPUID.OAH: EAX[15:8] > 1
		2	EN_PMC2	If CPUID.OAH: EAX[15:8] > 2
		n	EN_PMCn	If CPUID.OAH: EAX[15:8] >
		31:n+1	Reserved	
		32	EN_FIXED_CTR0	If CPUID.OAH: EDX[4:0] > 0
		33	EN_FIXED_CTR1	If CPUID.OAH: EDX[4:0] > 1
		34	EN_FIXED_CTR2	If CPUID.OAH: EDX[4:0] > 2
		47:35	Reserved	
		48	EN_PERF_METRICS: If this bit is set and fixed counter 3 is effectively enabled, built-in performance metrics are enabled.	
		63:49	Reserved	
390H	912	IA32_PERF_GLOBAL_OVF_CTRL	Global Performance Counter Overflow Control (R/W)	If CPUID.OAH: EAX[7:0] > 0 && CPUID.OAH: EAX[7:0] <= 3
		0	Set 1 to Clear Ovf_PMCO bit.	If CPUID.OAH: EAX[15:8] > 0
		1	Set 1 to Clear Ovf_PMC1 bit.	If CPUID.OAH: EAX[15:8] > 1
		2	Set 1 to Clear Ovf_PMC2 bit.	If CPUID.OAH: EAX[15:8] > 2
		n	Set 1 to Clear Ovf_PMCn bit.	If CPUID.OAH: EAX[15:8] > n
		31:n	Reserved	
		32	Set 1 to Clear Ovf_FIXED_CTR0 bit.	If CPUID.OAH: EDX[4:0] > 0
		33	Set 1 to Clear Ovf_FIXED_CTR1 bit.	If CPUID.OAH: EDX[4:0] > 1
		34	Set 1 to Clear Ovf_FIXED_CTR2 bit.	If CPUID.OAH: EDX[4:0] > 2
		54:35	Reserved	
		55	Set 1 to Clear Trace_ToPA_PMI bit.	If (CPUID.(EAX=07H, ECX=0):EBX[25] = 1) && IA32_RTIT_CTL.ToPA = 1
		60:56	Reserved	

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		61	Set 1 to Clear Ovf_Uncore bit.	06_2EH
		62	Set 1 to Clear OvfBuf bit.	If CPUID.OAH: EAX[7:0] > 0
		63	Set 1 to clear CondChgd bit.	If CPUID.OAH: EAX[7:0] > 0
390H	912	IA32_PERF_GLOBAL_STATUS_RESET	Global Performance Counter Overflow Reset Control (R/W)	If CPUID.0AH: EAX[7:0] > 3 II (CPUID.(EAX=07H, ECX=0):EBX[25] = 1 && CPUID.(EAX=014H, ECX=0):ECX[0] = 1)
		0	Set 1 to Clear Ovf_PMCO bit.	If CPUID.OAH: EAX[15:8] > 0
		1	Set 1 to Clear Ovf_PMC1 bit.	If CPUID.OAH: EAX[15:8] > 1
		2	Set 1 to Clear Ovf_PMC2 bit.	If CPUID.OAH: EAX[15:8] > 2
		n	Set 1 to Clear Ovf_PMCn bit.	If CPUID.OAH: EAX[15:8] > n
		31:n	Reserved	
		32	Set 1 to Clear Ovf_FIXED_CTR0 bit.	If CPUID.OAH: EDX[4:0] > 0
		33	Set 1 to Clear Ovf_FIXED_CTR1 bit.	If CPUID.OAH: EDX[4:0] > 1
		34	Set 1 to Clear Ovf_FIXED_CTR2 bit.	If CPUID.OAH: EDX[4:0] > 2
		47:35	Reserved	
		48	RESET_OVF_PERF_METRICS: If this bit is set, it will clear the status bit in the IA32_PERF_GLOBAL_STATUS register for the PERF_METRICS counters.	
		54:49	Reserved	
		55	Set 1 to Clear Trace_ToPA_PMI bit.	If CPUID.(EAX=07H, ECX=0):EBX[25] = 1 && CPUID.(EAX=014H, ECX=0):ECX[0] = 1
		57:56	Reserved	
		58	Set 1 to Clear LBR_Frz bit.	If CPUID.OAH: EAX[7:0] > 3
		59	Set 1 to Clear CTR_Frz bit.	If CPUID.OAH: EAX[7:0] > 3
		58	Set 1 to Clear ASCI bit.	If CPUID.OAH: EAX[7:0] > 3
		61	Set 1 to Clear Ovf_Uncore bit.	06_2EH
		62	Set 1 to Clear OvfBuf bit.	If CPUID.OAH: EAX[7:0] > 0
		63	Set 1 to clear CondChgd bit.	If CPUID.OAH: EAX[7:0] > 0
391H	913	IA32_PERF_GLOBAL_STATUS_SET	Global Performance Counter Overflow Set Control (R/W)	If CPUID.0AH: EAX[7:0] > 3 II (CPUID.(EAX=07H, ECX=0):EBX[25] = 1 && CPUID.(EAX=014H, ECX=0):ECX[0] = 1)

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		0	Set 1 to cause Ovf_PMC0 = 1.	If CPUID.OAH: EAX[7:0] > 3
		1	Set 1 to cause Ovf_PMC1 = 1.	If CPUID.OAH: EAX[15:8] > 1
		2	Set 1 to cause Ovf_PMC2 = 1.	If CPUID.OAH: EAX[15:8] > 2
		n	Set 1 to cause Ovf_PMCn = 1.	If CPUID.OAH: EAX[15:8] > n
		31:n	Reserved	
		32	Set 1 to cause Ovf_FIXED_CTR0 = 1.	If CPUID.OAH: EAX[7:0] > 3
		33	Set 1 to cause Ovf_FIXED_CTR1 = 1.	If CPUID.OAH: EAX[7:0] > 3
		34	Set 1 to cause Ovf_FIXED_CTR2 = 1.	If CPUID.OAH: EAX[7:0] > 3
		47:35	Reserved	
		48	SET_OVF_PERF_METRICS: If this bit is set, it will set the status bit in the IA32_PERF_GLOBAL_STATUS register for the PERF_METRICS counters.	
		54:49	Reserved	
		55	Set 1 to cause Trace_ToPA_PMI = 1.	If CPUID.(EAX=07H, ECX=0):EBX[25] = 1 && CPUID.(EAX=014H, ECX=0):ECX[0] = 1
		57:56	Reserved	
		58	Set 1 to cause LBR_Frz = 1.	If CPUID.OAH: EAX[7:0] > 3
		59	Set 1 to cause CTR_Frz = 1.	If CPUID.OAH: EAX[7:0] > 3
		58	Set 1 to cause ASCI = 1.	If CPUID.OAH: EAX[7:0] > 3
		61	Set 1 to cause Ovf_Uncore = 1.	If CPUID.OAH: EAX[7:0] > 3
		62	Set 1 to cause OvfBuf = 1.	If CPUID.OAH: EAX[7:0] > 3
		63	Reserved	
392H	914	IA32_PERF_GLOBAL_INUSE	Indicator that core perfmon interface is in use. (R/O)	If CPUID.OAH: EAX[7:0] > 3
		0	IA32_PERFEVTSEL0 in use.	
		1	IA32_PERFEVTSEL1 in use.	If CPUID.OAH: EAX[15:8] > 1
		2	IA32_PERFEVTSEL2 in use.	If CPUID.OAH: EAX[15:8] > 2
		n	IA32_PERFEVTSELn in use.	If CPUID.OAH: EAX[15:8] > n
		31:n+1	Reserved	
		32	IA32_FIXED_CTR0 in use.	
		33	IA32_FIXED_CTR1 in use.	
		34	IA32_FIXED_CTR2 in use.	

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)		Comment
Hex	Decimal			
		62:35	Reserved or model specific.	
		63	PMI in use.	
3F1H	1009	IA32_PEBS_ENABLE	PEBS Control (R/W)	
		0	Enable PEBS on IA32_PMC0.	06_0FH
		3:1	Reserved or model specific.	
		31:4	Reserved	
		35:32	Reserved or model specific.	
		63:36	Reserved	
400H	1024	IA32_MC0_CTL	MCO_CTL	If IA32_MCG_CAP.CNT >0
401H	1025	IA32_MC0_STATUS	MCO_STATUS	If IA32_MCG_CAP.CNT >0
402H	1026	IA32_MCO_ADDR ¹	MCO_ADDR	If IA32_MCG_CAP.CNT >0
403H	1027	IA32_MC0_MISC	MCO_MISC	If IA32_MCG_CAP.CNT >0
404H	1028	IA32_MC1_CTL	MC1_CTL	If IA32_MCG_CAP.CNT >1
405H	1029	IA32_MC1_STATUS	MC1_STATUS	If IA32_MCG_CAP.CNT >1
406H	1030	IA32_MC1_ADDR ²	MC1_ADDR	If IA32_MCG_CAP.CNT >1
407H	1031	IA32_MC1_MISC	MC1_MISC	If IA32_MCG_CAP.CNT >1
408H	1032	IA32_MC2_CTL	MC2_CTL	If IA32_MCG_CAP.CNT >2
409H	1033	IA32_MC2_STATUS	MC2_STATUS	If IA32_MCG_CAP.CNT >2
40AH	1034	IA32_MC2_ADDR ¹	MC2_ADDR	If IA32_MCG_CAP.CNT >2
40BH	1035	IA32_MC2_MISC	MC2_MISC	If IA32_MCG_CAP.CNT >2
40CH	1036	IA32_MC3_CTL	MC3_CTL	If IA32_MCG_CAP.CNT >3
40DH	1037	IA32_MC3_STATUS	MC3_STATUS	If IA32_MCG_CAP.CNT >3
40EH	1038	IA32_MC3_ADDR ¹	MC3_ADDR	If IA32_MCG_CAP.CNT >3
40FH	1039	IA32_MC3_MISC	MC3_MISC	If IA32_MCG_CAP.CNT >3
410H	1040	IA32_MC4_CTL	MC4_CTL	If IA32_MCG_CAP.CNT >4
411H	1041	IA32_MC4_STATUS	MC4_STATUS	If IA32_MCG_CAP.CNT >4
412H	1042	IA32_MC4_ADDR ¹	MC4_ADDR	If IA32_MCG_CAP.CNT >4
413H	1043	IA32_MC4_MISC	MC4_MISC	If IA32_MCG_CAP.CNT >4
414H	1044	IA32_MC5_CTL	MC5_CTL	If IA32_MCG_CAP.CNT >5
415H	1045	IA32_MC5_STATUS	MC5_STATUS	If IA32_MCG_CAP.CNT >5
416H	1046	IA32_MC5_ADDR ¹	MC5_ADDR	If IA32_MCG_CAP.CNT >5
417H	1047	IA32_MC5_MISC	MC5_MISC	If IA32_MCG_CAP.CNT >5
418H	1048	IA32_MC6_CTL	MC6_CTL	If IA32_MCG_CAP.CNT >6
419H	1049	IA32_MC6_STATUS	MC6_STATUS	If IA32_MCG_CAP.CNT >6
41AH	1050	IA32_MC6_ADDR ¹	MC6_ADDR	If IA32_MCG_CAP.CNT >6
41BH	1051	IA32_MC6_MISC	MC6_MISC	If IA32_MCG_CAP.CNT >6
41CH	1052	IA32_MC7_CTL	MC7_CTL	If IA32_MCG_CAP.CNT >7

Table 2-2. IA-32 Architectural MSRs (Contd.)

	jister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
41DH	1053	IA32_MC7_STATUS	MC7_STATUS	If IA32_MCG_CAP.CNT >7
41EH	1054	IA32_MC7_ADDR ¹	MC7_ADDR	If IA32_MCG_CAP.CNT >7
41FH	1055	IA32_MC7_MISC	MC7_MISC	If IA32_MCG_CAP.CNT >7
420H	1056	IA32_MC8_CTL	MC8_CTL	If IA32_MCG_CAP.CNT >8
421H	1057	IA32_MC8_STATUS	MC8_STATUS	If IA32_MCG_CAP.CNT >8
422H	1058	IA32_MC8_ADDR ¹	MC8_ADDR	If IA32_MCG_CAP.CNT >8
423H	1059	IA32_MC8_MISC	MC8_MISC	If IA32_MCG_CAP.CNT >8
424H	1060	IA32_MC9_CTL	MC9_CTL	If IA32_MCG_CAP.CNT >9
425H	1061	IA32_MC9_STATUS	MC9_STATUS	If IA32_MCG_CAP.CNT >9
426H	1062	IA32_MC9_ADDR ¹	MC9_ADDR	If IA32_MCG_CAP.CNT >9
427H	1063	IA32_MC9_MISC	MC9_MISC	If IA32_MCG_CAP.CNT >9
428H	1064	IA32_MC10_CTL	MC10_CTL	If IA32_MCG_CAP.CNT >10
429H	1065	IA32_MC10_STATUS	MC10_STATUS	If IA32_MCG_CAP.CNT >10
42AH	1066	IA32_MC10_ADDR ¹	MC10_ADDR	If IA32_MCG_CAP.CNT >10
42BH	1067	IA32_MC10_MISC	MC10_MISC	If IA32_MCG_CAP.CNT >10
42CH	1068	IA32_MC11_CTL	MC11_CTL	If IA32_MCG_CAP.CNT >11
42DH	1069	IA32_MC11_STATUS	MC11_STATUS	If IA32_MCG_CAP.CNT >11
42EH	1070	IA32_MC11_ADDR ¹	MC11_ADDR	If IA32_MCG_CAP.CNT >11
42FH	1071	IA32_MC11_MISC	MC11_MISC	If IA32_MCG_CAP.CNT >11
430H	1072	IA32_MC12_CTL	MC12_CTL	If IA32_MCG_CAP.CNT >12
431H	1073	IA32_MC12_STATUS	MC12_STATUS	If IA32_MCG_CAP.CNT >12
432H	1074	IA32_MC12_ADDR ¹	MC12_ADDR	If IA32_MCG_CAP.CNT >12
433H	1075	IA32_MC12_MISC	MC12_MISC	If IA32_MCG_CAP.CNT >12
434H	1076	IA32_MC13_CTL	MC13_CTL	If IA32_MCG_CAP.CNT >13
435H	1077	IA32_MC13_STATUS	MC13_STATUS	If IA32_MCG_CAP.CNT >13
436H	1078	IA32_MC13_ADDR ¹	MC13_ADDR	If IA32_MCG_CAP.CNT >13
437H	1079	IA32_MC13_MISC	MC13_MISC	If IA32_MCG_CAP.CNT >13
438H	1080	IA32_MC14_CTL	MC14_CTL	If IA32_MCG_CAP.CNT >14
439H	1081	IA32_MC14_STATUS	MC14_STATUS	If IA32_MCG_CAP.CNT >14
43AH	1082	IA32_MC14_ADDR ¹	MC14_ADDR	If IA32_MCG_CAP.CNT >14
43BH	1083	IA32_MC14_MISC	MC14_MISC	If IA32_MCG_CAP.CNT >14
43CH	1084	IA32_MC15_CTL	MC15_CTL	If IA32_MCG_CAP.CNT >15
43DH	1085	IA32_MC15_STATUS	MC15_STATUS	If IA32_MCG_CAP.CNT >15
43EH	1086	IA32_MC15_ADDR ¹	MC15_ADDR	If IA32_MCG_CAP.CNT >15
43FH	1087	IA32_MC15_MISC	MC15_MISC	If IA32_MCG_CAP.CNT >15
440H	1088	IA32_MC16_CTL	MC16_CTL	If IA32_MCG_CAP.CNT >16
441H	1089	IA32_MC16_STATUS	MC16_STATUS	If IA32_MCG_CAP.CNT >16

Table 2-2. IA-32 Architectural MSRs (Contd.)

	jister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal	(Former Fish Nume)	r isiv bit bescription	Comment
442H	1090	IA32_MC16_ADDR ¹	MC16_ADDR	If IA32_MCG_CAP.CNT >16
443H	1091	IA32_MC16_MISC	MC16_MISC	If IA32_MCG_CAP.CNT >16
444H	1092	IA32_MC17_CTL	MC17_CTL	If IA32_MCG_CAP.CNT >17
445H	1093	IA32_MC17_STATUS	MC17_STATUS	If IA32_MCG_CAP.CNT >17
446H	1094	IA32_MC17_ADDR ¹	MC17_ADDR	If IA32_MCG_CAP.CNT >17
447H	1095	IA32_MC17_MISC	MC17_MISC	If IA32_MCG_CAP.CNT >17
448H	1096	IA32_MC18_CTL	MC18_CTL	If IA32_MCG_CAP.CNT >18
449H	1097	IA32_MC18_STATUS	MC18_STATUS	If IA32_MCG_CAP.CNT >18
44AH	1098	IA32_MC18_ADDR ¹	MC18_ADDR	If IA32_MCG_CAP.CNT >18
44BH	1099	IA32_MC18_MISC	MC18_MISC	If IA32_MCG_CAP.CNT >18
44CH	1100	IA32_MC19_CTL	MC19_CTL	If IA32_MCG_CAP.CNT >19
44DH	1101	IA32_MC19_STATUS	MC19_STATUS	If IA32_MCG_CAP.CNT >19
44EH	1102	IA32_MC19_ADDR ¹	MC19_ADDR	If IA32_MCG_CAP.CNT >19
44FH	1103	IA32_MC19_MISC	MC19_MISC	If IA32_MCG_CAP.CNT >19
450H	1104	IA32_MC20_CTL	MC20_CTL	If IA32_MCG_CAP.CNT >20
451H	1105	IA32_MC20_STATUS	MC20_STATUS	If IA32_MCG_CAP.CNT >20
452H	1106	IA32_MC20_ADDR ¹	MC20_ADDR	If IA32_MCG_CAP.CNT >20
453H	1107	IA32_MC20_MISC	MC20_MISC	If IA32_MCG_CAP.CNT >20
454H	1108	IA32_MC21_CTL	MC21_CTL	If IA32_MCG_CAP.CNT >21
455H	1109	IA32_MC21_STATUS	MC21_STATUS	If IA32_MCG_CAP.CNT >21
456H	1110	IA32_MC21_ADDR ¹	MC21_ADDR	If IA32_MCG_CAP.CNT >21
457H	1111	IA32_MC21_MISC	MC21_MISC	If IA32_MCG_CAP.CNT >21
458H	1112	IA32_MC22_CTL	MC22_CTL	If IA32_MCG_CAP.CNT >22
459H	1113	IA32_MC22_STATUS	MC22_STATUS	If IA32_MCG_CAP.CNT >22
45AH	1114	IA32_MC22_ADDR ¹	MC22_ADDR	If IA32_MCG_CAP.CNT >22
45BH	1115	IA32_MC22_MISC	MC22_MISC	If IA32_MCG_CAP.CNT >22
45CH	1116	IA32_MC23_CTL	MC23_CTL	If IA32_MCG_CAP.CNT >23
45DH	1117	IA32_MC23_STATUS	MC23_STATUS	If IA32_MCG_CAP.CNT >23
45EH	1118	IA32_MC23_ADDR ¹	MC23_ADDR	If IA32_MCG_CAP.CNT >23
45FH	1119	IA32_MC23_MISC	MC23_MISC	If IA32_MCG_CAP.CNT >23
460H	1120	IA32_MC24_CTL	MC24_CTL	If IA32_MCG_CAP.CNT >24
461H	1121	IA32_MC24_STATUS	MC24_STATUS	If IA32_MCG_CAP.CNT >24
462H	1122	IA32_MC24_ADDR ¹	MC24_ADDR	If IA32_MCG_CAP.CNT >24
463H	1123	IA32_MC24_MISC	MC24_MISC	If IA32_MCG_CAP.CNT >24
464H	1124	IA32_MC25_CTL	MC25_CTL	If IA32_MCG_CAP.CNT >25
465H	1125	IA32_MC25_STATUS	MC25_STATUS	If IA32_MCG_CAP.CNT >25
466H	1126	IA32_MC25_ADDR ¹	MC25_ADDR	If IA32_MCG_CAP.CNT >25

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
467H	1127	IA32_MC25_MISC	MC25_MISC	If IA32_MCG_CAP.CNT >25
468H	1128	IA32_MC26_CTL	MC26_CTL	If IA32_MCG_CAP.CNT >26
469H	1129	IA32_MC26_STATUS	MC26_STATUS	If IA32_MCG_CAP.CNT >26
46AH	1130	IA32_MC26_ADDR ¹	MC26_ADDR	If IA32_MCG_CAP.CNT >26
46BH	1131	IA32_MC26_MISC	MC26_MISC	If IA32_MCG_CAP.CNT >26
46CH	1132	IA32_MC27_CTL	MC27_CTL	If IA32_MCG_CAP.CNT >27
46DH	1133	IA32_MC27_STATUS	MC27_STATUS	If IA32_MCG_CAP.CNT >27
46EH	1134	IA32_MC27_ADDR ¹	MC27_ADDR	If IA32_MCG_CAP.CNT >27
46FH	1135	IA32_MC27_MISC	MC27_MISC	If IA32_MCG_CAP.CNT >27
470H	1136	IA32_MC28_CTL	MC28_CTL	If IA32_MCG_CAP.CNT >28
471H	1137	IA32_MC28_STATUS	MC28_STATUS	If IA32_MCG_CAP.CNT >28
472H	1138	IA32_MC28_ADDR ¹	MC28_ADDR	If IA32_MCG_CAP.CNT >28
473H	1139	IA32_MC28_MISC	MC28_MISC	If IA32_MCG_CAP.CNT >28
480H	1152	IA32_VMX_BASIC	Reporting Register of Basic VMX Capabilities (R/O)	If CPUID.01H:ECX.[5] = 1
			See Appendix A.1, "Basic VMX Information."	
481H	1153	ia32_vmx_pinbased_ctls	Capability Reporting Register of Pin-Based VM-Execution Controls (R/O)	If CPUID.01H:ECX.[5] = 1
			See Appendix A.3.1, "Pin-Based VM- Execution Controls."	
482H	1154	IA32_VMX_PROCBASED_CTLS	Capability Reporting Register of Primary Processor-Based VM-Execution Controls (R/O)	If CPUID.01H:ECX.[5] = 1
			See Appendix A.3.2, "Primary Processor- Based VM-Execution Controls."	
483H	1155	IA32_VMX_EXIT_CTLS	Capability Reporting Register of Primary VM-Exit Controls (R/O)	If CPUID.01H:ECX.[5] = 1
			See Appendix A.4.1, "Primary VM-Exit Controls."	
484H	1156	IA32_VMX_ENTRY_CTLS	Capability Reporting Register of VM-Entry Controls (R/O)	If CPUID.01H:ECX.[5] = 1
			See Appendix A.5, "VM-Entry Controls."	
485H	1157	IA32_VMX_MISC	Reporting Register of Miscellaneous VMX Capabilities (R/O)	If CPUID.01H:ECX.[5] = 1
			See Appendix A.6, "Miscellaneous Data."	
486H	1158	IA32_VMX_CR0_FIXED0	Capability Reporting Register of CRO Bits Fixed to 0 (R/O)	If CPUID.01H:ECX.[5] = 1
			See Appendix A.7, "VMX-Fixed Bits in CR0."	
487H	1159	IA32_VMX_CRO_FIXED1	Capability Reporting Register of CRO Bits Fixed to 1 (R/O)	If CPUID.01H:ECX.[5] = 1
			See Appendix A.7, "VMX-Fixed Bits in CR0."	

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
488H	1160	IA32_VMX_CR4_FIXED0	Capability Reporting Register of CR4 Bits Fixed to 0 (R/0) See Appendix A.8, "VMX-Fixed Bits in CR4."	If CPUID.01H:ECX.[5] = 1
489H	1161	IA32 VMV CD4 CIVCD1	Capability Reporting Register of CR4 Bits	If CPUID.01H:ECX.[5] = 1
409FI	1101	IA32_VMX_CR4_FIXED1	Fixed to 1 (R/0)	II CPOID.OTH.ECX.[5] - T
40.011	1162	LADD VANY VANCE CALLINA	See Appendix A.8, "VMX-Fixed Bits in CR4."	If CDUID 0411 CCV [E] = 1
48AH	1162	IA32_VMX_VMCS_ENUM	Capability Reporting Register of VMCS Field Enumeration (R/O) See Appendix A.9, "VMCS Enumeration."	If CPUID.01H:ECX.[5] = 1
40011	1160	LARRA VANV. BROCERASER, STUSP		14 (CD) 11D 0411 CC) (FE1 0.0
48BH	1163	IA32_VMX_PROCBASED_CTLS2	Capability Reporting Register of Secondary Processor-Based VM-Execution Controls (R/O) See Appendix A.3.3, "Secondary Processor-	If (CPUID.01H:ECX.[5] && IA32_VMX_PROCBASED_C TLS[63])
			Based VM-Execution Controls."	
48CH	1164	IA32_VMX_EPT_VPID_CAP	Capability Reporting Register of EPT and VPID (R/O)	If (CPUID.01H:ECX.[5] && IA32_VMX_PROCBASED_C
			See Appendix A.10, "VPID and EPT Capabilities."	TLS[63] && (IA32_VMX_PROCBASED_C TLS2[33] IA32_VMX_PROCBASED_C TLS2[37]))
48DH	1165	IA32_VMX_TRUE_PINBASED_CTLS	Capability Reporting Register of Pin-Based VM-Execution Flex Controls (R/O)	If (CPUID.01H:ECX.[5] && IA32_VMX_BASIC[55])
			See Appendix A.3.1, "Pin-Based VM- Execution Controls."	
48EH	1166	IA32_VMX_TRUE_PROCBASED_CTLS	Capability Reporting Register of Primary Processor-Based VM-Execution Flex Controls (R/O)	If(CPUID.01H:ECX.[5] && IA32_VMX_BASIC[55])
			See Appendix A.3.2, "Primary Processor- Based VM-Execution Controls."	
48FH	1167	IA32_VMX_TRUE_EXIT_CTLS	Capability Reporting Register of VM-Exit Flex Controls (R/O)	If(CPUID.01H:ECX.[5] && IA32_VMX_BASIC[55])
			See Appendix A.4, "VM-Exit Controls."	
490H	1168	IA32_VMX_TRUE_ENTRY_CTLS	Capability Reporting Register of VM-Entry Flex Controls (R/O)	If(CPUID.01H:ECX.[5] && IA32_VMX_BASIC[55])
			See Appendix A.5, "VM-Entry Controls."	
491H	1169	IA32_VMX_VMFUNC	Capability Reporting Register of VM- Function Controls (R/O)	If(CPUID.01H:ECX.[5] && IA32_VMX_PROCBASED_C TLS[63] && IA32_VMX_PROCBASED_C TLS2[45])
492H	1170	IA32_VMX_PROCBASED_CTLS3	Capability Reporting Register of Tertiary Processor-Based VM-Execution Controls (R/O)	If (CPUID.01H:ECX.[5] && IA32_VMX_PROCBASED_C TLS[49])
			See Appendix A.3.4, "Tertiary Processor- Based VM-Execution Controls."	

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal		_	
493H	1171	IA32_VMX_EXIT_CTLS2	Capability Reporting Register of Secondary VM-Exit Controls (R/O)	If (CPUID.01H:ECX.[5] && IA32_VMX_EXIT_CTLS[63]
			See Appendix A.4.2, "Secondary VM-Exit Controls.")
4C1H	1217	IA32_A_PMCO	Full Width Writable IA32_PMCO Alias (R/W)	(If CPUID.OAH: EAX[15:8] > 0) &&
				IA32_PERF_CAPABILITIES[13] = 1
4C2H	1218	IA32_A_PMC1	Full Width Writable IA32_PMC1 Alias (R/W)	(If CPUID.OAH: EAX[15:8] > 1) &&
				IA32_PERF_CAPABILITIES[13] = 1
4C3H	1219	IA32_A_PMC2	Full Width Writable IA32_PMC2 Alias (R/W)	(If CPUID.OAH: EAX[15:8] > 2) &&
				IA32_PERF_CAPABILITIES[13] = 1
4C4H	1220	IA32_A_PMC3	Full Width Writable IA32_PMC3 Alias (R/W)	(If CPUID.OAH: EAX[15:8] > 3) &&
				IA32_PERF_CAPABILITIES[13] = 1
4C5H	1221	IA32_A_PMC4	Full Width Writable IA32_PMC4 Alias (R/W)	(If CPUID.OAH: EAX[15:8] > 4) &&
				IA32_PERF_CAPABILITIES[13] = 1
4C6H	1222	IA32_A_PMC5	Full Width Writable IA32_PMC5 Alias (R/W)	(If CPUID.OAH: EAX[15:8] > 5) &&
				IA32_PERF_CAPABILITIES[13] = 1
4C7H	1223	IA32_A_PMC6	Full Width Writable IA32_PMC6 Alias (R/W)	(If CPUID.OAH: EAX[15:8] > 6) &&
				IA32_PERF_CAPABILITIES[13] = 1
4C8H	1224	IA32_A_PMC7	Full Width Writable IA32_PMC7 Alias (R/W)	(If CPUID.OAH: EAX[15:8] > 7) &&
				IA32_PERF_CAPABILITIES[13] = 1
4D0H	1232	IA32_MCG_EXT_CTL	Allows software to signal some MCEs to only a single logical processor in the system. (R/W)	If IA32_MCG_CAP.LMCE_P =1
			See Section 16.3.1.4, "IA32_MCG_EXT_CTL MSR".	
		0	LMCE_EN	
		63:1	Reserved	
500H	1280	IA32_SGX_SVN_STATUS	Status and SVN Threshold of SGX Support for ACM (R/O).	If CPUID.(EAX=07H, ECX=0H): EBX[2] = 1

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		0	Lock	See Section 38.11.3, "Interactions with Authenticated Code Modules (ACMs)".
		15:1	Reserved	
		23:16	SGX_SVN_SINIT	See Section 38.11.3, "Interactions with Authenticated Code Modules (ACMs)".
		63:24	Reserved	
560H	1376	IA32_RTIT_OUTPUT_BASE	Trace Output Base Register (R/W)	If ((CPUID.(EAX=07H, ECX=0):EBX[25] = 1) && ((CPUID.(EAX=14H,ECX=0):E CX[0] = 1) (CPUID.(EAX=14H,ECX=0):E CX[2] = 1)))
		6:0	Reserved	
		MAXPHYADDR ⁴ -1:7	Base physical address.	
		63:MAXPHYADDR	Reserved	
561H	1377	IA32_RTIT_OUTPUT_MASK_PTRS	Trace Output Mask Pointers Register (R/W)	If ((CPUID.(EAX=07H, ECX=0):EBX[25] = 1) && ((CPUID.(EAX=14H,ECX=0):E CX[0] = 1) (CPUID.(EAX=14H,ECX=0):E CX[2] = 1)))
		6:0	Reserved	
		31:7	MaskOrTableOffset	
		63:32	Output Offset	
570H	1392	IA32_RTIT_CTL	Trace Control Register (R/W)	If (CPUID.(EAX=07H, ECX=0):EBX[25] = 1)
		0	TraceEn	
		1	CYCEn	If (CPUID.(EAX=07H, ECX=0):EBX[1] = 1)
		2	OS	
		3	User	
		4	PwrEvtEn	If (CPUID.(EAX=07H, ECX=1):EBX[5] = 1)
		5	FUPonPTW	If (CPUID.(EAX=07H, ECX=1):EBX[4] = 1)
		6	FabricEn	If (CPUID.(EAX=07H, ECX=0):ECX[3] = 1)
		7	CR3Filter	If (CPUID.(EAX=14H, ECX=0):EBX[0] = 1)
		8	ToPA	

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		9	MTCEn	If (CPUID.(EAX=07H, ECX=0):EBX[3] = 1)
		10	TSCEn	
		11	DisRETC	
		12	PTWEn	If (CPUID.(EAX=07H, ECX=1):EBX[4] = 1)
		13	BranchEn	
		17:14	MTCFreq	If (CPUID.(EAX=07H, ECX=0):EBX[3] = 1)
		18	Reserved, must be zero.	
		22:19	CycThresh	If (CPUID.(EAX=07H, ECX=0):EBX[1] = 1)
		23	Reserved, must be zero.	
		27:24	PSBFreq	If (CPUID.(EAX=07H, ECX=0):EBX[1] = 1)
		30:28	Reserved, must be zero.	
		31	EventEn	If (CPUID.(EAX=14H, ECX=0):EBX[7] = 1)
		35:32	ADDRO_CFG	If (CPUID.(EAX=07H, ECX=1):EAX[2:0] > 0)
		39:36	ADDR1_CFG	If (CPUID.(EAX=07H, ECX=1):EAX[2:0] > 1)
		43:40	ADDR2_CFG	If (CPUID.(EAX=07H, ECX=1):EAX[2:0] > 2)
		47:44	ADDR3_CFG	If (CPUID.(EAX=07H, ECX=1):EAX[2:0] > 3)
		54:48	Reserved, must be zero.	
		55	DisTNT	If (CPUID.(EAX=14H, ECX=0):EBX[8] = 1)
		56	InjectPsbPmiOnEnable	If (CPUID.(EAX=07H, ECX=1):EBX[6] = 1)
		63:57	Reserved, must be zero.	
571H	1393	IA32_RTIT_STATUS	Tracing Status Register (R/W)	If (CPUID.(EAX=07H, ECX=0):EBX[25] = 1)
		0	FilterEn (writes ignored)	If (CPUID.(EAX=07H, ECX=0):EBX[2] = 1)
		1	ContexEn (writes ignored)	
		2	TriggerEn (writes ignored)	
		3	Reserved	
		4	Error	
		5	Stopped	

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		6	PendPSB	If (CPUID.(EAX=07H, ECX=0):EBX[6] = 1)
		7	PendToPAPMI	If (CPUID.(EAX=07H, ECX=0):EBX[6] = 1)
		31:8	Reserved, must be zero.	
		48:32	PacketByteCnt	If (CPUID.(EAX=07H, ECX=0):EBX[1] > 3)
		63:49	Reserved	
572H	1394	IA32_RTIT_CR3_MATCH	Trace Filter CR3 Match Register (R/W)	If (CPUID.(EAX=07H, ECX=0):EBX[25] = 1)
		4:0	Reserved	
		63:5	CR3[63:5] value to match.	
580H	1408	IA32_RTIT_ADDRO_A	Region 0 Start Address (R/W)	If (CPUID.(EAX=07H, ECX=1):EAX[2:0] > 0)
		47:0	Virtual Address	
		63:48	SignExt_VA	
581H	1409	IA32_RTIT_ADDRO_B	Region 0 End Address (R/W)	If (CPUID.(EAX=07H, ECX=1):EAX[2:0] > 0)
		47:0	Virtual Address	
		63:48	SignExt_VA	
582H	1410	IA32_RTIT_ADDR1_A	Region 1 Start Address (R/W)	If (CPUID.(EAX=07H, ECX=1):EAX[2:0] > 1)
		47:0	Virtual Address	
		63:48	SignExt_VA	
583H	1411	IA32_RTIT_ADDR1_B	Region 1 End Address (R/W)	If (CPUID.(EAX=07H, ECX=1):EAX[2:0] > 1)
		47:0	Virtual Address	
		63:48	SignExt_VA	
584H	1412	IA32_RTIT_ADDR2_A	Region 2 Start Address (R/W)	If (CPUID.(EAX=07H, ECX=1):EAX[2:0] > 2)
		47:0	Virtual Address	
		63:48	SignExt_VA	
585H	1413	IA32_RTIT_ADDR2_B	Region 2 End Address (R/W)	If (CPUID.(EAX=07H, ECX=1):EAX[2:0] > 2)
		47:0	Virtual Address	
		63:48	SignExt_VA	
586H	1414	IA32_RTIT_ADDR3_A	Region 3 Start Address (R/W)	If (CPUID.(EAX=07H, ECX=1):EAX[2:0] > 3)
		47:0	Virtual Address	
		63:48	SignExt_VA	

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
587H	1415	IA32_RTIT_ADDR3_B	Region 3 End Address (R/W)	If (CPUID.(EAX=07H, ECX=1):EAX[2:0] > 3)
		47:0	Virtual Address	
		63:48	SignExt_VA	
600H	1536	IA32_DS_AREA	DS Save Area (R/W) Points to the linear address of the first byte of the DS buffer management area, which is used to manage the BTS and PEBS buffers. See Section 20.6.3.4, "Debug Store (DS) Mechanism."	If(CPUID.01H:EDX.DS[21] = 1
		63:0	The linear address of the first byte of the DS buffer management area, if IA-32e mode is active.	
		31:0	The linear address of the first byte of the DS buffer management area, if not in IA-32e mode.	
		63:32	Reserved if not in IA-32e mode.	
6A0H	1696	IA32_U_CET	Configure User Mode CET (R/W)	Bits 1:0 are defined if CPUID.(EAX=07H, ECX=0H):ECX.CET_SS[07] = 1. Bits 5:2 and bits 63:10 are defined if CPUID.(EAX=07H, ECX=0H):EDX.CET_IBT[20] = 1.
		0	SH_STK_EN: When set to 1, enable shadow stacks at CPL3.	
	2 3	1	WR_SHSTK_EN: When set to 1, enables the WRSSD/WRSSQ instructions.	
		2	ENDBR_EN: When set to 1, enables indirect branch tracking.	
		3	LEG_IW_EN: Enable legacy compatibility treatment for indirect branch tracking.	
		4	NO_TRACK_EN: When set to 1, enables use of no-track prefix for indirect branch tracking.	
		5	SUPPRESS_DIS: When set to 1, disables suppression of CET indirect branch tracking on legacy compatibility.	
		9:6	Reserved; must be zero.	
		10	SUPPRESS: When set to 1, indirect branch tracking is suppressed. This bit can be written to 1 only if TRACKER is written as IDLE.	_

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		11	TRACKER: Value of the indirect branch tracking state machine. Values: IDLE (0), WAIT_FOR_ENDBRANCH(1).	
		63:12	EB_LEG_BITMAP_BASE: Linear address bits 63:12 of a legacy code page bitmap used for legacy compatibility when indirect branch tracking is enabled.	
			If the processor does not support Intel 64 architecture, these fields have only 32 bits; bits 63:32 of the MSRs are reserved. On processors that support Intel 64 architecture this value cannot represent a non-canonical address. In protected mode, only 31:0 are loaded. The linear address written must be aligned to 8 bytes and bits 2:0 must be 0 (hardware requires bits 1:0 to be 0).	
6A2H	1698	IA32_S_CET	Configure Supervisor Mode CET (R/W)	See IA32_U_CET (6A0H) for reference; similar format.
6A4H	1700	IA32_PL0_SSP	Linear address to be loaded into SSP on transition to privilege level 0. (R/W) If the processor does not support Intel 64 architecture, these fields have only 32 bits; bits 63:32 of the MSRs are reserved. On processors that support Intel 64 architecture this value cannot represent a non-canonical address. In protected mode, only 31:0 are loaded. The linear address written must be aligned to 8 bytes and bits 2:0 must be 0 (hardware requires bits 1:0 to be 0).	If CPUID.(EAX=07H, ECX=0H):ECX.CET_SS[07] = 1
6A5H	1701	IA32_PL1_SSP	Linear address to be loaded into SSP on transition to privilege level 1. (R/W) If the processor does not support Intel 64 architecture, these fields have only 32 bits; bits 63:32 of the MSRs are reserved. On processors that support Intel 64 architecture this value cannot represent a non-canonical address. In protected mode, only 31:0 are loaded. The linear address written must be aligned to 8 bytes and bits 2:0 must be 0 (hardware requires bits 1:0 to be 0).	If CPUID.(EAX=07H, ECX=0H):ECX.CET_SS[07] = 1

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
6A6H	1702	IA32_PL2_SSP	Linear address to be loaded into SSP on transition to privilege level 2. (R/W)	If CPUID.(EAX=07H, ECX=0H):ECX.CET_SS[07]
			If the processor does not support Intel 64 architecture, these fields have only 32 bits; bits 63:32 of the MSRs are reserved. On processors that support Intel 64 architecture this value cannot represent a non-canonical address. In protected mode, only 31:0 are loaded. The linear address written must be aligned to 8 bytes and bits 2:0 must be 0 (hardware requires bits 1:0 to be 0).	= 1
6A7H	1703	IA32_PL3_SSP	Linear address to be loaded into SSP on transition to privilege level 3. (R/W) If the processor does not support Intel 64 architecture, these fields have only 32 bits; bits 63:32 of the MSRs are reserved. On processors that support Intel 64 architecture this value cannot represent a non-canonical address. In protected mode, only 31:0 are loaded. The linear address written must be aligned to 8 bytes and bits 2:0 must be 0 (hardware requires bits 1:0 to be 0).	If CPUID.(EAX=07H, ECX=0H):ECX.CET_SS[07] = 1
6A8H	1704	IA32_INTERRUPT_SSP_TABLE_ADDR	Linear address of a table of seven shadow stack pointers that are selected in IA-32e mode using the IST index (when not 0) from the interrupt gate descriptor. (R/W) This MSR is not present on processors that do not support Intel 64 architecture. This field cannot represent a non-canonical address.	If CPUID.(EAX=07H, ECX=0H):ECX.CET_SS[07] = 1
6E0H	1760	IA32_TSC_DEADLINE	TSC Target of Local APIC's TSC Deadline Mode (R/W)	If CPUID.01H:ECX.[24] = 1
6E1H	1761	IA32_PKRS	Specifies the PK permissions associated with each protection domain for supervisor pages (R/W)	If CPUID.(EAX=07H, ECX=0H):ECX.PKS [31] = 1
		31:0	For domain i (i between 0 and 15), bits 2i and 2i+1 contain the AD and WD permissions, respectively.	
		63:32	Reserved.	
770H	1904	IA32_PM_ENABLE	Enable/disable HWP (R/W)	If CPUID.06H:EAX.[7] = 1
		0	HWP_ENABLE (R/W1-Once) See Section 15.4.2, "Enabling HWP".	If CPUID.06H:EAX.[7] = 1
		63:1	Reserved	
771H	1905	IA32_HWP_CAPABILITIES	HWP Performance Range Enumeration (R/O)	If CPUID.06H:EAX.[7] = 1

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		7:0	Highest_Performance See Section 15.4.3, "HWP Performance Range and Dynamic Capabilities".	If CPUID.06H:EAX.[7] = 1
		15:8	Guaranteed_Performance See Section 15.4.3, "HWP Performance Range and Dynamic Capabilities".	If CPUID.06H:EAX.[7] = 1
		23:16	Most_Efficient_Performance See Section 15.4.3, "HWP Performance Range and Dynamic Capabilities".	If CPUID.06H:EAX.[7] = 1
		31:24	Lowest_Performance See Section 15.4.3, "HWP Performance Range and Dynamic Capabilities".	If CPUID.06H:EAX.[7] = 1
		63:32	Reserved	
772H	1906	IA32_HWP_REQUEST_PKG	Power Management Control Hints for All Logical Processors in a Package (R/W)	If CPUID.06H:EAX.[11] = 1
		7:0	Minimum_Performance See Section 15.4.4, "Managing HWP".	If CPUID.06H:EAX.[11] = 1
		15:8	Maximum_Performance See Section 15.4.4, "Managing HWP".	If CPUID.06H:EAX.[11] = 1
		23:16	Desired_Performance See Section 15.4.4, "Managing HWP".	If CPUID.06H:EAX.[11] = 1
		31:24	Energy_Performance_Preference See Section 15.4.4, "Managing HWP".	If CPUID.06H:EAX.[11] = 1 && CPUID.06H:EAX.[10] = 1
		41:32	Activity_Window See Section 15.4.4, "Managing HWP".	If CPUID.06H:EAX.[11] = 1 && CPUID.06H:EAX.[9] = 1
		63:42	Reserved	
773H	1907	IA32_HWP_INTERRUPT	Control HWP Native Interrupts (R/W)	If CPUID.06H:EAX.[8] = 1
		0	EN_Guaranteed_Performance_Change See Section 15.4.6, "HWP Notifications".	If CPUID.06H:EAX.[8] = 1
		1	EN_Excursion_Minimum See Section 15.4.6, "HWP Notifications".	If CPUID.06H:EAX.[8] = 1
		63:2	Reserved	
774H	1908	IA32_HWP_REQUEST	Power Management Control Hints to a Logical Processor (R/W)	If CPUID.06H:EAX.[7] = 1
		7:0	Minimum_Performance See Section 15.4.4, "Managing HWP".	If CPUID.06H:EAX.[7] = 1
		15:8	Maximum_Performance See Section 15.4.4, "Managing HWP".	If CPUID.06H:EAX.[7] = 1

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		23:16	Desired_Performance See Section 15.4.4, "Managing HWP".	If CPUID.06H:EAX.[7] = 1
		31:24	Energy_Performance_Preference See Section 15.4.4, "Managing HWP".	If CPUID.06H:EAX.[7] = 1 && CPUID.06H:EAX.[10] = 1
		41:32	Activity_Window See Section 15.4.4, "Managing HWP".	If CPUID.06H:EAX.[7] = 1 && CPUID.06H:EAX.[9] = 1
		42	Package_Control See Section 15.4.4, "Managing HWP".	If CPUID.06H:EAX.[7] = 1 && CPUID.06H:EAX.[11] = 1
		63:43	Reserved	
775H	1909	IA32_PECI_HWP_REQUEST_INFO	IA32_PECI_HWP_REQUEST_INFO	
		7:0	Minimum Performance (MINIMUM_PERFORMANCE): Used by OS to read the latest value of PECI minimum performance input. Default value is 0.	
		15:8	Maximum Performance (MAXIMUM_PERFORMANCE): Used by OS to read the latest value of PECI maximum performance input. Default value is 0.	
		23:16	Reserved.	
		31:24	Energy Performance Preference (ENERGY_PERFORMANCE_PREFERENCE): Used by OS to read the latest value of PECI Energy Performance Preference input. Default value is O.	
		59:32	Reserved.	
		60	EPP PECI Override (EPP_PECI_OVERRIDE): Indicates whether PECI is currently overriding the Energy Performance Preference input. If set to '1', PECI is overriding the Energy Performance Preference input. If clear (0), OS has control over Energy Performance Preference input. Default value is 0.	
		61	Reserved.	
		62	Max PECI Override (MAX_PECI_OVERRIDE):	
			Indicates whether PECI is currently overriding the Maximum Performance input. If set to '1', PECI is overriding the Maximum Performance input. If clear (0), OS has control over Maximum Performance input. Default value is 0.	

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	+	MSR/Bit Description	Comment
Hex	Decimal			
		63	Min PECI Override (MIN_PECI_OVERRIDE): Indicates whether PECI is currently overriding the Minimum Performance input. If set to '1', PECI is overriding the Minimum Performance input. If clear (0), OS has control over Minimum Performance input.	
			Default value is 0.	
776H	1910	IA32_HWP_CTL	IA32_HWP_CTL	If CPUID.06H:EAX.[22] = 1
		0	PKG_CTL_POLARITY Defines which HWP Request MSR is used whether Thread level or package level. When package MSR is used, the thread MSR valid bits define which thread MSR fields override the package. Default value is 0.	If CPUID.06H:EAX.[22] = 1
		63:1	Reserved	
777H	1911	IA32_HWP_STATUS	Log bits indicating changes to Guaranteed & excursions to Minimum (R/W)	If CPUID.06H:EAX.[7] = 1
		0	Guaranteed_Performance_Change (R/WC0) See Section 15.4.5, "HWP Feedback".	If CPUID.06H:EAX.[7] = 1
		1	Reserved	
		2	Excursion_To_Minimum (R/WC0) See Section 15.4.5, "HWP Feedback".	If CPUID.06H:EAX.[7] = 1
		63:3	Reserved	
802H	2050	IA32_X2APIC_APICID	x2APIC ID Register (R/O) See x2APIC Specification.	If CPUID.01H:ECX[21] = 1 && IA32_APIC_BASE.[10] = 1
803H	2051	IA32_X2APIC_VERSION	x2APIC Version Register (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
808H	2056	IA32_X2APIC_TPR	x2APIC Task Priority Register (R/W)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
80AH	2058	IA32_X2APIC_PPR	x2APIC Processor Priority Register (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
80BH	2059	IA32_X2APIC_EOI	x2APIC EOI Register (W/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
80DH	2061	IA32_X2APIC_LDR	x2APIC Logical Destination Register (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
80FH	2063	IA32_X2APIC_SIVR	x2APIC Spurious Interrupt Vector Register (R/W)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
810H	2064	IA32_X2APIC_ISRO	x2APIC In-Service Register Bits 31:0 (R/0)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
811H	2065	IA32_X2APIC_ISR1	x2APIC In-Service Register Bits 63:32 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
812H	2066	IA32_X2APIC_ISR2	x2APIC In-Service Register Bits 95:64 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
813H	2067	IA32_X2APIC_ISR3	x2APIC In-Service Register Bits 127:96 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
814H	2068	IA32_X2APIC_ISR4	x2APIC In-Service Register Bits 159:128 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
815H	2069	IA32_X2APIC_ISR5	x2APIC In-Service Register Bits 191:160 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
816H	2070	IA32_X2APIC_ISR6	x2APIC In-Service Register Bits 223:192 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
817H	2071	IA32_X2APIC_ISR7	x2APIC In-Service Register Bits 255:224 (R/O)	If CPUID.01H:ECX.[21] = 1 &&IA32_APIC_BASE.[10] = 1
818H	2072	IA32_X2APIC_TMR0	x2APIC Trigger Mode Register Bits 31:0 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
819H	2073	IA32_X2APIC_TMR1	x2APIC Trigger Mode Register Bits 63:32 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
81AH	2074	IA32_X2APIC_TMR2	x2APIC Trigger Mode Register Bits 95:64 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
81BH	2075	IA32_X2APIC_TMR3	x2APIC Trigger Mode Register Bits 127:96 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
81CH	2076	IA32_X2APIC_TMR4	x2APIC Trigger Mode Register Bits 159:128 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
81DH	2077	IA32_X2APIC_TMR5	x2APIC Trigger Mode Register Bits 191:160 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
81EH	2078	IA32_X2APIC_TMR6	x2APIC Trigger Mode Register Bits 223:192 (R/O)	If (CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1)

Table 2-2. IA-32 Architectural MSRs (Contd.)

	jister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
81FH	2079	IA32_X2APIC_TMR7	x2APIC Trigger Mode Register Bits 255:224 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
820H	2080	IA32_X2APIC_IRRO	x2APIC Interrupt Request Register Bits 31:0 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
821H	2081	IA32_X2APIC_IRR1	x2APIC Interrupt Request Register Bits 63:32 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
822H	2082	IA32_X2APIC_IRR2	x2APIC Interrupt Request Register Bits 95:64 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
823H	2083	IA32_X2APIC_IRR3	x2APIC Interrupt Request Register Bits 127:96 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
824H	2084	IA32_X2APIC_IRR4	x2APIC Interrupt Request Register Bits 159:128 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
825H	2085	IA32_X2APIC_IRR5	x2APIC Interrupt Request Register Bits 191:160 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
826H	2086	IA32_X2APIC_IRR6	x2APIC Interrupt Request Register Bits 223:192 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
827H	2087	IA32_X2APIC_IRR7	x2APIC Interrupt Request Register Bits 255:224 (R/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
828H	2088	IA32_X2APIC_ESR	x2APIC Error Status Register (R/W)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
82FH	2095	IA32_X2APIC_LVT_CMCI	x2APIC LVT Corrected Machine Check Interrupt Register (R/W)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
830H	2096	IA32_X2APIC_ICR	x2APIC Interrupt Command Register (R/W)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
832H	2098	IA32_X2APIC_LVT_TIMER	x2APIC LVT Timer Interrupt Register (R/W)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
833H	2099	IA32_X2APIC_LVT_THERMAL	x2APIC LVT Thermal Sensor Interrupt Register (R/W)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
834H	2100	IA32_X2APIC_LVT_PMI	x2APIC LVT Performance Monitor Interrupt Register (R/W)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
835H	2101	IA32_X2APIC_LVT_LINTO	x2APIC LVT LINTO Register (R/W)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
836H	2102	IA32_X2APIC_LVT_LINT1	x2APIC LVT LINT1 Register (R/W)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
837H	2103	IA32_X2APIC_LVT_ERROR	x2APIC LVT Error Register (R/W)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
838H	2104	IA32_X2APIC_INIT_COUNT	x2APIC Initial Count Register (R/W)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
839H	2105	IA32_X2APIC_CUR_COUNT	x2APIC Current Count Register (R/0)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
83EH	2110	IA32_X2APIC_DIV_CONF	x2APIC Divide Configuration Register (R/W)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
83FH	2111	IA32_X2APIC_SELF_IPI	x2APIC Self IPI Register (W/O)	If CPUID.01H:ECX.[21] = 1 && IA32_APIC_BASE.[10] = 1
981H	2433	IA32_TME_CAPABILITY	Memory Encryption Capability MSR	If CPUID.07H:ECX.[13] = 1
		0	Support for AES-XTS 128-bit encryption algorithm. (NIST standard)	
		1	Support for AES-XTS 128-bit encryption with integrity algorithm.	
		2	Support for AES-XTS 256-bit encryption algorithm.	
		30:3	Reserved.	
		31	TME encryption bypass supported.	
		35:32	MK_TME_MAX_KEYID_BITS	
			Number of bits which can be allocated for usage as key identifiers for multi-key memory encryption.	
			4 bits allow for a maximum value of 15, which could address 32K keys.	
			Zero if MKTME is not supported.	
		50:36	MK_TME_MAX_KEYS	
			Indicates the maximum number of keys which are available for usage.	
			This value may not be a power of 2.	
			KeyID 0 is specially reserved and is not accounted for in this field.	
		63:51	Reserved.	

Table 2-2. IA-32 Architectural MSRs (Contd.)

_	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal	-		
982H	2434	IA32_TME_ACTIVATE	Memory Encryption Activation MSR This MSR is used to lock the MSRs listed below. Any write to the following MSRs will be ignored after they are locked. The lock is reset when CPU is reset. • IA32_TME_ACTIVATE	If CPUID.07H:ECX.[13] = 1
			■ IA32_TME_EXCLUDE_MASK	
			• IA32_TME_EXCLUDE_BASE	
			Note that IA32_TME_EXCLUDE_MASK and IA32_TME_EXCLUDE_BASE must be configured before IA32_TME_ACTIVATE.	
		0	Lock R/O - Will be set upon successful WRMSR (or first SMI); written value ignored.	
		1	Hardware Encryption Enable This bit also enables MKTME; MKTME cannot be enabled without enabling encryption hardware.	
		2	Key Select	
			0: Create a new TME key (expected cold/warm boot).	
			1: Restore the TME key from storage (Expected when resume from standby).	
		3	Save TME Key for Standby	
			Save key into storage to be used when resume from standby.	
			Note: This may not be supported in all processors.	
		7:4	TME Policy/Encryption Algorithm	
			Only algorithms enumerated in IA32_TME_CAPABILITY are allowed.	
			For example:	
			0000 - AES-XTS-128.	
			0001 - AES-XTS-128 with integrity.	
			0010 - AES-XTS-256.	
			Other values are invalid.	
		30:8	Reserved.	
				·

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		31	TME Encryption Bypass Enable	
			When encryption hardware is enabled:	
			 Total Memory Encryption is enabled using a CPU generated ephemeral key based on a hardware random number generator when this bit is set to 0. Total Memory Encryption is bypassed (no encryption/decryption for KeyIDO) when this bit is set to 1. Software must inspect Hardware Encryption Enable (bit 1) and TME encryption bypass Enable (bit 31) to determine if TME encryption is enabled. 	
		35:32	MK_TME_KEYID_BITS Reserved if MKTME is not enumerated,	
			otherwise: The number of key identifier bits to allocate to MKTME usage. Similar to enumeration, this is an encoded value.	
			Writing a value greater than MK_TME_MAX_KEYID_BITS will result in #GP.	
			Writing a non-zero value to this field will #GP if bit 1 of EAX (Hardware Encryption Enable) is not also set to '1, as encryption hardware must be enabled to use MKTME.	
			Example: To support 255 keys, this field would be set to a value of 8.	
		47:36	Reserved.	
		63:48	MK_TME_CRYPTO_ALGS	
			Reserved if MKTME is not enumerated, otherwise:	
			Bit 48: AES-XTS 128.	
			Bit 49: AES-XTS 128 with integrity.	
			Bit 50: AES-XTS 256. Bit 63:51: Reserved (#GP)	
			Bit 03.31. Reserved (#dr) Bitmask for BIOS to set which encryption algorithms are allowed for MKTME, would be later enforced by the key loading ISA ('1 = allowed).	
983H	2435	IA32_TME_EXCLUDE_MASK	Memory Encryption Exclude Mask	If CPUID.07H:ECX.[13] = 1
		10:0	Reserved.	
		11	Enable: When set to '1', then TME_EXCLUDE_BASE and TME_EXCLUDE_MASK are used to define an exclusion region for TME/MKTME (for KeyID=0).	

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		MAXPHYSADDR-1:12	TMEEMASK: This field indicates the bits that must match TMEEBASE in order to qualify as a TME/MKTME (for KeyID=0) exclusion memory range access.	
		63:MAXPHYSADDR	Reserved; must be zero.	
984H	2436	IA32_TME_EXCLUDE_BASE	Memory Encryption Exclude Base	IF CPUID.07H:ECX.[13] = 1
		11:0	Reserved.	
		MAXPHYSADDR-1:12	TMEEBASE: Base physical address to be excluded for TME/MKTME (for KeyID=0) encryption.	
		63:MAXPHYSADDR	Reserved; must be zero.	
985H	2437	IA32_UINTR_RR	User Interrupt Request Register (R/W)	IF CPUID.07H.01H:EDX[13]=1
		63:0	UIRR Bitmap of requested user interrupt vectors.	
986H	2438	IA32_UINTR_HANDLER	User Interrupt Handler Address (R/W)	IF CPUID.07H.01H:EDX[13]=1
		63:0	UIHANDLER	
			User interrupt handler linear address.	
987H	2439	IA32_UINTR_STACKADJUST	User Interrupt Stack Adjustment (R/W)	IF CPUID.07H.01H:EDX[13]=1
		0	LOAD_RSP	
			User interrupt stack mode.	
		2:1	Reserved.	
		63:3	STACK_ADJUST	
			Stack adjust value.	
988H	2440	IA32_UINTR_MISC	User-Interrupt Target-Table Size and Notification Vector (R/W)	If CPUID.07H.01H:EDX[13]=1
		31:0	UITTSZ The highest index of a valid entry in the user-interrupt target table. Valid entries are indices 0UITTSZ (inclusive).	
		39:32	UINV	
			User-interrupt notification vector.	
		63:40	Reserved.	
989H	2441	IA32_UINTR_PD	User Interrupt PID Address (R/W)	If CPUID.07H.01H:EDX[13]=1
		5:0	Reserved.	
		63:6	UPIDADDR	
			User-interrupt notification processing accesses a UPID at this linear address.	

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name) MSR/Bit Description	Comment	
Hex	Decimal			
98AH	2442	IA32_UINTR_TT User-Interrupt Target Table (R/W)		If CPUID.07H.01H:EDX[13]=1
		0	SENDUIPI_ENABLE	
			User-interrupt target table is valid.	
		3:1	Reserved.	
		63:4	UITTADDR	
			User-interrupt target table base linear address.	
990H	2448	IA32_COPY_STATUS ⁵	Status of Most Recent Platform to Local or Local to Platform Copies (R/O)	If ((CPUID.19H:EBX[4] = 1) && (CPUID.(07H,0).ECX[23] = 1))
		0	IWKEY_COPY_SUCCESSFUL	If ((CPUID.19H:EBX[4] = 1)
			Status of most recent copy to or from IWKeyBackup	&& (CPUID.(07H,0).ECX[23] = 1))
		63:1	Reserved	
991H	2449	IA32_IWKEYBACKUP_STATUS ⁵	Information about IWKeyBackup Register (R/O)	If ((CPUID.19H:EBX[4] = 1) && (CPUID.(07H,0).ECX[23] =1))
		0	Backup/Restore Valid	IF ((CPUID.19H:EBX[4] = 1)
			Cleared when a write to IWKeyBackup is initiated, and then set when the latest write of IWKeyBackup has been written to storage that persists across S3/S4 sleep state. If S3/S4 is entered between when an IWKeyBackup write occurs and when this bit is set, then IWKeyBackup may not be recovered after S3/S4 exit. During S3/S4 sleep state exit (system wake up), this bit is cleared. It is set again when IWKeyBackup is restored from persistent storage and thus available to be copied to IWKey using IA32_COPY_PLATFORM_TO_LOCAL MSR. Another write to IWKeyBackup (via IA32_COPY_LOCAL_TO_PLATFORM MSR) may fail if a previous write has not yet set this bit.	&& (CPUID.(07H,0).ECX[23] =1))
		1	Reserved	
		2	Backup Key Storage Read/Write Error Updated prior to backup/restore valid being set. Set when an error is encountered while backing up or restoring a key to persistent storage.	IF ((CPUID.19H:EBX[4] = 1) && (CPUID.(07H,0).ECX[23] =1))

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment	
Hex	Decimal				
		3	IWKeyBackup Consumed Set after the previous backup operation has been consumed by the platform. This does not indicate that the system is ready for a second IWKeyBackup write as the previous IWKeyBackup write may still need to set Backup/restore valid.	IF ((CPUID.19H:EBX[4] = 1) && (CPUID.(07H,0).ECX[23] =1))	
		63:4	Reserved		
C80H	3200	IA32_DEBUG_INTERFACE	Silicon Debug Feature Control (R/W)	If CPUID.01H:ECX.[11] = 1	
		0	Enable (R/W) BIOS set 1 to enable Silicon debug features. Default is 0.	If CPUID.01H:ECX.[11] = 1	
		29:1	Reserved		
		30	Lock (R/W): If 1, locks any further change to the MSR. The lock bit is set automatically on the first SMI assertion even if not explicitly set by BIOS. Default is 0.	If CPUID.01H:ECX.[11] = 1	
		31	Debug Occurred (R/O): This "sticky bit" is set by hardware to indicate the status of bit 0. Default is 0.	If CPUID.01H:ECX.[11] = 1	
		63:32	Reserved		
C81H	3201	IA32_L3_QOS_CFG	L3 QOS Configuration (R/W)	If (CPUID.(EAX=10H, ECX=1):ECX.[2] = 1)	
		0	Enable (R/W) Set 1 to enable L3 CAT masks and COS to operate in Code and Data Prioritization (CDP) mode.		
		63:1	Reserved. Attempts to write to reserved bits result in a #GP(0).		
C82H	3202	IA32_L2_QOS_CFG	L2 QOS Configuration (R/W)	If (CPUID.(EAX=10H, ECX=2):ECX.[2] = 1)	
		0	Enable (R/W) Set 1 to enable L2 CAT masks and COS to operate in Code and Data Prioritization (CDP) mode.		
		63:1	Reserved. Attempts to write to reserved bits result in a #GP(0).		
C8DH	3213	IA32_QM_EVTSEL	Monitoring Event Select Register (R/W)	If (CPUID.(EAX=07H, ECX=0):EBX.[12] = 1)	
		7:0	Event ID: ID of a supported monitoring event to report via IA32_QM_CTR.		
		31:8	Reserved		

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		N+31:32	Resource Monitoring ID: ID for monitoring hardware to report monitored data via IA32_QM_CTR.	N = Ceil (Log ₂ (CPUID.(EAX= 0FH, ECX=0H).EBX[31:0] +1))
		63:N+32	Reserved	
C8EH	3214	IA32_QM_CTR	Monitoring Counter Register (R/O)	If (CPUID.(EAX=07H, ECX=0):EBX.[12] = 1)
		61:0	Resource Monitored Data	
		62	Unavailable: If 1, indicates data for this RMID is not available or not monitored for this resource or RMID.	
		63	Error: If 1, indicates an unsupported RMID or event type was written to IA32_PQR_QM_EVTSEL.	
C8FH	C8FH 3215 IA32_PQR_ASSOC Resource Association Register (F		Resource Association Register (R/W)	If ((CPUID.(EAX=07H, ECX=0):EBX[12] =1) or (CPUID.(EAX=07H, ECX=0):EBX[15] =1))
		N-1:0	Resource Monitoring ID (R/W): ID for monitoring hardware to track internal operation, e.g., memory access.	N = Ceil (Log ₂ (CPUID.(EAX= 0FH, ECX=0H).EBX[31:0] +1))
		31:N	Reserved	
		63:32	COS (R/W): The class of service (COS) to enforce (on writes); returns the current COS when read.	If (CPUID.(EAX=07H, ECX=0):EBX.[15] = 1)
C90H - D8FH	3216 - 3471	Reserved MSR Address Space for CAT Mask Registers	See Section 18.19.4.1, "Enumeration and Detection Support of Cache Allocation Technology".	
C90H	3216	IA32_L3_MASK_0	L3 CAT Mask for COSO (R/W)	If (CPUID.(EAX=10H, ECX=0H):EBX[1]!= 0)
		31:0	Capacity Bit Mask (R/W)	
		63:32	Reserved	
C90H+	3216+n	IA32_L3_MASK_n	L3 CAT Mask for COSn (R/W)	n = CPUID.(EAX=10H, ECX=1H):EDX[15:0]
		31:0	Capacity Bit Mask (R/W)	
		63:32	Reserved	
D10H - D4FH	3344 - 3407	Reserved MSR Address Space for L2 CAT Mask Registers	See Section 18.19.4.1, "Enumeration and Detection Support of Cache Allocation Technology".	

Table 2-2. IA-32 Architectural MSRs (Contd.)

	jister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment	
Hex	Decimal				
D10H 3344		IA32_L2_MASK_0	L2 CAT Mask for COSO (R/W)	If (CPUID.(EAX=10H, ECX=0H):EBX[2]!= 0)	
		31:0	Capacity Bit Mask (R/W)		
		63:32	Reserved		
D10H+ n	3344+n	IA32_L2_MASK_n	L2 CAT Mask for COSn (R/W)	n = CPUID.(EAX=10H, ECX=2H):EDX[15:0]	
		31:0	Capacity Bit Mask (R/W)		
		63:32	Reserved		
D90H	3472	IA32_BNDCFGS	Supervisor State of MPX Configuration (R/W)	If (CPUID.(EAX=07H, ECX=0H):EBX[14] = 1)	
		0	EN: Enable Intel MPX in supervisor mode.		
		1	BNDPRESERVE: Preserve the bounds registers for near branch instructions in the absence of the BND prefix.		
		11:2	Reserved, must be zero.		
		63:12	Base Address of Bound Directory.		
D91H	3473	IA32_COPY_LOCAL_TO_PLATFORM ⁵	Copy Local State to Platform State (W)	IF ((CPUID.19H:EBX[4] = 1) && (CPUID.(EAX=07H, ECX=0H).ECX[23] = 1))	
		0	IWKeyBackup Copy IWKey to IWKeyBackup	IF ((CPUID.19H:EBX[4] = 1) && (CPUID.(EAX=07H, ECX=0H).ECX[23] = 1))	
		63:1	Reserved		
D92H	3474	IA32_COPY_PLATFORM_TO_LOCAL ⁵	Copy Platform State to Local State (W)	IF ((CPUID.19H:EBX[4] = 1) && (CPUID.(EAX=07H, ECX=0H).ECX[23] = 1))	
		0	IWKeyBackup Copy IWKeyBackup to IWKey	IF ((CPUID.19H:EBX[4] = 1) && (CPUID.(EAX=07H, ECX=0H).ECX[23] = 1))	
		63:1	Reserved		
D93H	3475	IA32_PASID	Process Address Space Identifier (R/W)		
		19:0	Process address space identifier (PASID). Specifies the PASID of the currently running software thread.		
		30:20	Reserved		
		31	Valid. Execution of ENQCMD causes a #GP if this bit is clear.		
		63:32	Reserved		

Table 2-2. IA-32 Architectural MSRs (Contd.)

	jister Iress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
DAOH	3488	IA32_XSS	Extended Supervisor State Mask (R/W)	If(CPUID.(ODH, 1):EAX.[3] = 1
		7:0	Reserved.	
		8	PT State (R/W)	
		9	Reserved.	
		10	PASID State (R/W)	
		11	CET_U State (R/W)	
		12	CET_S State (R/W)	
		13	HDC State (R/W)	
		14	UINTR State (R/W)	
		15	LBR State (R/W)	
		16	HWP State (R/W)	
		63:17	Reserved.	
DBOH	3504	IA32_PKG_HDC_CTL	Package Level Enable/disable HDC (R/W)	If CPUID.06H:EAX.[13] = 1
		0	HDC_Pkg_Enable (R/W) Force HDC idling or wake up HDC-idled logical processors in the package. See Section 15.5.2, "Package level Enabling HDC".	If CPUID.06H:EAX.[13] = 1
		63:1	Reserved	
DB1H	3505	IA32_PM_CTL1	Enable/disable HWP (R/W)	If CPUID.06H:EAX.[13] = 1
		0	HDC_Allow_Block (R/W) Allow/Block this logical processor for package level HDC control. See Section 15.5.3.	
		63:1	Reserved	
DB2H	3506	IA32_THREAD_STALL	Per-Logical_Processor HDC Idle Residency (R/0)	If CPUID.06H:EAX.[13] = 1
		63:0	Stall_Cycle_Cnt (R/W) Stalled cycles due to HDC forced idle on this logical processor. See Section 15.5.4.1.	If CPUID.06H:EAX.[13] = 1
1200H -	4608 -	IA32_LBR_x_INFO	Last Branch Record Entry X Info Register (R/W)	
121FH	4639		An attempt to read or write IA32_LBR_x_INFO such that $x \ge IA32_LBR_DEPTH.DEPTH$ will #GP.	
		15:0	CYC_CNT The elapsed CPU cycles (saturating) since the last LBR was recorded. See Section 18.1.3.3.	Reset Value: 0

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment	
Hex	Decimal				
		55:16	Undefined, may be zero or non-zero. Writes of non- zero values do not fault, but reads may return a different value.	Reset Value: 0	
		59:56	BR_TYPE The branch type recorded by this LBR. Encodings: 0000B: COND 0001B: JMP Indirect 0010B: JMP Direct 0011B: CALL Indirect	Reset Value: 0	
			0101B: RET 011xB: Reserved 1xxxB: Other Branch		
		60	CYC_CNT_VALID CYC_CNT value is valid. See Section 19.1.3.3.	Reset Value: 0	
		61	TSX_ABORT This LBR record is a TSX abort. On processors that do not support Intel TSX (CPUID.07H.EBX.HLE[bit 4]=0 and CPUID.07H.EBX.RTM[bit 11]=0), this bit is undefined.	Reset Value: 0	
		62	IN_TSX This LBR record records a branch that retired during a TSX transaction. On processors that do not support Intel TSX (CPUID.07H.EBX.HLE[bit 4]=0 and CPUID.07H.EBX.RTM[bit 11]=0), this bit is undefined.	Reset Value: 0	
		63	MISPRED The recorded branch direction (conditional branch) or target (indirect branch) was mispredicted.	Reset Value: 0	
1406H	5126	IA32_MCU_CONTROL	MCU Control (R/W) Controls the behavior of the Microcode Update Trigger MSR, IA32_BIOS_UPDT_TRIG.	If CPUID.07H.0H:EDX[29]=1 && MSR.IA32_ARCH_CAPABILI TIES.MCU_CONTROL=1	
		0	LOCK Once set, further writes to this MSR will cause a #GP(0) fault. Bypassed during SMM if EN_SMM_BYPASS (bit 2) is set.		

Table 2-2. IA-32 Architectural MSRs (Contd.)

	gister dress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		1	DIS_MCU_LOAD If this bit is set on a given logical processor, then any subsequent attempts to load a microcode update by that logical processor will be silently dropped (WRMSR 0x79 has no effect).	
		2	EN_SMM_BYPASS If set, then writes to IA32_MCU_CONTROL are allowed during SMM regardless of the LOCK bit. This enables BIOS to Opt-In to the SMM Bypass functionality.	
		63:3	Reserved.	
14CEH	5326	IA32_LBR_CTL	Last Branch Record Enabling and Configuration Register (R/W)	
		0	LBREn When set, enables LBR recording.	Reset Value: 0
		1	OS When set, allows LBR recording when CPL == 0.	Reset Value: 0
		2	USR When set, allows LBR recording when CPL != 0.	Reset Value: 0
		3	CALL_STACK When set, records branches in call-stack mode. See Section 19.1.2.4.	Reset Value: 0
		15:4	Reserved	Reset Value: 0
		16	COND When set, records taken conditional branches. See Section 19.1.2.3.	
		17	NEAR_REL_JMP When set, records near relative JMPs. See Section 19.1.2.3.	
		18	NEAR_IND_JMP When set, records near indirect JMPs. See Section 19.1.2.3.	
		19	NEAR_REL_CALL When set, records near relative CALLs. See Section 19.1.2.3.	
		20	NEAR_IND_CALL When set, records near indirect CALLs. See Section 19.1.2.3.	
		21	NEAR_RET When set, records near RETs. See Section 19.1.2.3.	

Table 2-2. IA-32 Architectural MSRs (Contd.)

	jister Iress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		22	OTHER_BRANCH When set, records other branches. See Section 19.1.2.3.	
		63:23	Reserved	
14CFH	5327	IA32_LBR_DEPTH	Last Branch Record Maximum Stack Depth Register (R/W)	
		N:0	DEPTH The number of LBRs to be used for recording. Supported values are indicated by the bitmap in CPUID.(EAX=01CH,ECX=0):EAX[7:0]. The reset value will match the maximum supported by the CPU. Writes of unsupported values will #GP fault.	Reset Value: Varies
		63:N+1	Reserved	Reset Value: 0
1500H - 151FH	register (R/W). An attempt to read or write IA32_LBR_x_FROM_IP such that $x \ge$		register (R/W). An attempt to read or write	
		63:0	FROM_IP The source IP of the recorded branch or event, in canonical form. Writes to bits above MAXLINADDR-1 are ignored.	Reset Value: 0
1600H - 161FH	5632 - 5663	IA32_LBR_x_TO_IP	Last Branch Record Entry X Destination IP Register (R/W) An attempt to read or write IA32_LBR_x_TO_IP such that x ≥ IA32_LBR_DEPTH.DEPTH will #GP.	
		63:0	TO_IP The destination IP of the recorded branch or event, in canonical form. Writes to bits above MAXLINADDR-1 are ignored.	Reset Value: 0
17D0H	6096	IA32_HW_FEEDBACK_PTR	Hardware Feedback Interface Pointer	If CPUID.06H:EAX.[19] = 1
		0	Valid (R/W) When set to 1, indicates a valid pointer is programmed into the ADDR field of the MSR.	
		11:1	Reserved	
		(MAXPHYADDR-1):12	ADDR (R/W) Physical address of the page frame of the first page of the hardware feedback interface structure.	
		63:MAXPHYADDR	Reserved	
17D1H	6097	IA32_HW_FEEDBACK_CONFIG	Hardware Feedback Interface Configuration	If CPUID.06H:EAX.[19] = 1

Table 2-2. IA-32 Architectural MSRs (Contd.)

	jister Iress	Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		0	Enable (R/W) When set to 1, enables the hardware feedback interface.	
		63:1	Reserved	
17D2H	6098	IA32_THREAD_FEEDBACK_CHAR	Thread Feedback Characteristics (R/O)	If CPUID.06H:EAX.[23] = 1
		7:0	Application Class ID, pointing into the Intel Thread Director structure.	
		62:8	Reserved	
		63	Valid bit. When set to 1 the OS Scheduler can use the Class ID (in bits 7:0) for its scheduling decisions. If this bit is 0, the Class ID field should be ignored. It is recommended that the OS uses the last known Class ID of the software thread for its scheduling decisions.	
17D4H	6100	IA32_HW_FEEDBACK_THREAD_CONFI G	Hardware Feedback Thread Configuration (R/W)	
		0	Enables Intel Thread Director. When set to 1, logical processor scope Intel Thread Director is enabled. Default is 0 (disabled).	
		63:1	Reserved	
17DAH	6106	IA32_HRESET_ENABLE	History Reset Enable (R/W)	
		0	Enable reset of the Intel Thread Director history.	
		31:1	Reserved for other capabilities that can be reset by the HRESET instruction.	
		63:32	Reserved	
1B01H	6913	IA32_UARCH_MISC_CTL	IA32_UARCH_MISC_CTL	If IA32_ARCH_CAPABILITIES[12]=1
		0	Data Operand Independent Timing Mode (DOITM)	if IA32_arch_capabilities[12]=1
		63:1	Reserved	
4000_ 0000H - 4000_ 00FFH		Reserved MSR Address Space	All existing and future processors will not implement MSRs in this range.	
0080H		IA32_EFER	Extended Feature Enables	If (CPUID.80000001H:EDX.[2 0] CPUID.80000001H:EDX.[2 9])

Table 2-2. IA-32 Architectural MSRs (Contd.)

Register Address		Architectural MSR Name / Bit Fields (Former MSR Name)	MSR/Bit Description	Comment
Hex	Decimal			
		0	SYSCALL Enable: IA32_EFER.SCE (R/W) Enables SYSCALL/SYSRET instructions in 64-bit mode.	
		7:1	Reserved	
		8	IA-32e Mode Enable: IA32_EFER.LME (R/W) Enables IA-32e mode operation.	
		9	Reserved	
		10	IA-32e Mode Active: IA32_EFER.LMA (R) Indicates IA-32e mode is active when set.	
		11	Execute Disable Bit Enable: IA32_EFER.NXE (R/W)	
		63:12	Reserved	
C000_ 0081H		IA32_STAR	System Call Target Address (R/W)	If CPUID.80000001:EDX.[29] = 1
C000_ 0082H		IA32_LSTAR	IA-32e Mode System Call Target Address (R/W) Target RIP for the called procedure when SYSCALL is executed in 64-bit mode.	If CPUID.80000001:EDX.[29] = 1
C000_ 0083H		IA32_CSTAR	IA-32e Mode System Call Target Address (R/W) Not used, as the SYSCALL instruction is not recognized in compatibility mode.	If CPUID.80000001:EDX.[29] = 1
C000_ 0084H		IA32_FMASK	System Call Flag Mask (R/W)	If CPUID.80000001:EDX.[29] = 1
C000_ 0100H		IA32_FS_BASE	Map of BASE Address of FS (R/W)	If CPUID.80000001:EDX.[29] = 1
C000_ 0101H		IA32_GS_BASE	Map of BASE Address of GS (R/W)	If CPUID.80000001:EDX.[29] = 1
C000_ 0102H		IA32_KERNEL_GS_BASE	Swap Target of BASE Address of GS (R/W)	If CPUID.80000001:EDX.[29] = 1
C000_ 0103H		IA32_TSC_AUX	Auxiliary TSC (R/W)	If CPUID.80000001H: EDX[27] = 1 or CPUID.(EAX=7,ECX=0):ECX[bit 22] = 1
		31:0	AUX: Auxiliary signature of TSC.	
		63:32	Reserved	

NOTES:

1. Some older processors may have supported this MSR as model-specific and do not enumerate it with CPUID.

- 2. In processors based on Intel NetBurst® microarchitecture, MSR addresses 180H-197H are supported, software must treat them as model-specific. Starting with Intel Core Duo processors, MSR addresses 180H-185H, 188H-197H are reserved.
- 3. The *_ADDR MSRs may or may not be present; this depends on flag settings in IA32_MC*i_*STATUS. See Section 16.3.2.3 and Section 16.3.2.4 for more information.
- 4. MAXPHYADDR is reported by CPUID.80000008H:EAX[7:0].
- 5. Further details on Key Locker and usage of this MSR can be found here:

https://software.intel.com/content/www/us/en/develop/download/intel-key-locker-specification.html

2.2 MSRS IN THE INTEL® CORE™ 2 PROCESSOR FAMILY

Table 2-3 lists model-specific registers (MSRs) for the Intel Core 2 processor family and for Intel Xeon processors based on Intel Core microarchitecture, architectural MSR addresses are also included in Table 2-3. These processors have a CPUID signature with DisplayFamily_DisplayModel of 06_0FH, see Table 2-1.

MSRs listed in Table 2-2 and Table 2-3 are also supported by processors based on the Enhanced Intel Core microarchitecture. Processors based on the Enhanced Intel Core microarchitecture have the CPUID signature DisplayFamily DisplayModel of 06 17H.

The column "Shared/Unique" applies to multi-core processors based on Intel Core microarchitecture. "Unique" means each processor core has a separate MSR, or a bit field in an MSR governs only a core independently. "Shared" means the MSR or the bit field in an MSR address governs the operation of both processor cores.

Table 2-3. MSRs in Processors Based on Intel® Core™ Microarchitecture

Regi Add		Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
OH	0	IA32_P5_MC_ADDR	Unique	See Section 2.23, "MSRs in Pentium Processors."
1H	1	IA32_P5_MC_TYPE	Unique	See Section 2.23, "MSRs in Pentium Processors."
6H	6	IA32_MONITOR_FILTER_SIZE	Unique	See Section 9.10.5, "Monitor/Mwait Address Range Determination." and Table 2-2.
10H	16	IA32_TIME_STAMP_COUNTER	Unique	See Section 18.17, "Time-Stamp Counter," and see Table 2-2.
17H	23	IA32_PLATFORM_ID	Shared	Platform ID (R) See Table 2-2.
17H	23	MSR_PLATFORM_ID	Shared	Model Specific Platform ID (R)
		7:0		Reserved
		12:8		Maximum Qualified Ratio (R)
				The maximum allowed bus ratio.
		49:13		Reserved
		52:50		See Table 2-2.
		63:53		Reserved
1BH	27	IA32_APIC_BASE	Unique	See Section 11.4.4, "Local APIC Status and Location" and Table 2-2.
2AH	42	MSR_EBL_CR_POWERON	Shared	Processor Hard Power-On Configuration (R/W)
				Enables and disables processor features; (R) indicates current processor configuration.
		0		Reserved

Table 2-3. MSRs in Processors Based on Intel® Core™ Microarchitecture (Contd.)

	ister ress	Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
		1		Data Error Checking Enable (R/W) 1 = Enabled; 0 = Disabled Note: Not all processors implement R/W.
		2		Response Error Checking Enable (R/W) 1 = Enabled; 0 = Disabled Note: Not all processor implements R/W.
		3		MCERR# Drive Enable (R/W) 1 = Enabled; 0 = Disabled Note: Not all processors implement R/W.
		4		Address Parity Enable (R/W) 1 = Enabled; 0 = Disabled Note: Not all processors implement R/W.
		5		Reserved
		6		Reserved
		7		BINIT# Driver Enable (R/W) 1 = Enabled; 0 = Disabled Note: Not all processors implement R/W.
		8		Output Tri-state Enabled (R/O) 1 = Enabled; O = Disabled
		9		Execute BIST (R/O) 1 = Enabled; O = Disabled
		10		MCERR# Observation Enabled (R/O) 1 = Enabled; O = Disabled
		11		Intel TXT Capable Chipset. (R/O) 1 = Present; O = Not Present
		12		BINIT# Observation Enabled (R/O) 1 = Enabled; O = Disabled
		13		Reserved
		14		1 MByte Power on Reset Vector (R/O) 1 = 1 MByte; 0 = 4 GBytes
		15		Reserved
		17:16		APIC Cluster ID (R/O)
		18		N/2 Non-Integer Bus Ratio (R/O) 0 = Integer ratio; 1 = Non-integer ratio
		19		Reserved
		21: 20		Symmetric Arbitration ID (R/O)
		26:22		Integer Bus Frequency Ratio (R/O)

Table 2-3. MSRs in Processors Based on Intel® Core™ Microarchitecture (Contd.)

	ister ress	Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
ЗАН	58	MSR_FEATURE_CONTROL	Unique	Control Features in Intel 64 Processor (R/W)
				See Table 2-2.
		3	Unique	SMRR Enable (R/WL)
				When this bit is set and the lock bit is set, this makes the SMRR_PHYS_BASE and SMRR_PHYS_MASK registers read visible and writeable while in SMM.
40H	64	MSR_LASTBRANCH_0_FROM_IP	Unique	Last Branch Record O From IP (R/W)
				One of four pairs of last branch record registers on the last branch record stack. The From_IP part of the stack contains pointers to the source instruction. See also:
				Last Branch Record Stack TOS at 1C9H.Section 18.5.
41H	65	MSR_LASTBRANCH_1_FROM_IP	Unique	Last Branch Record 1 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
42H	66	MSR_LASTBRANCH_2_FROM_IP	Unique	Last Branch Record 2 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
43H	67	MSR_LASTBRANCH_3_FROM_IP	Unique	Last Branch Record 3 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
60H	96	MSR_LASTBRANCH_0_TO_IP	Unique	Last Branch Record 0 To IP (R/W)
				One of four pairs of last branch record registers on the last branch record stack. This To_IP part of the stack contains pointers to the destination instruction.
61H	97	MSR_LASTBRANCH_1_TO_IP	Unique	Last Branch Record 1 To IP (R/W)
				See description of MSR_LASTBRANCH_0_T0_IP.
62H	98	MSR_LASTBRANCH_2_TO_IP	Unique	Last Branch Record 2 To IP (R/W)
				See description of MSR_LASTBRANCH_0_T0_IP.
63H	99	MSR_LASTBRANCH_3_TO_IP	Unique	Last Branch Record 3 To IP (R/W)
				See description of MSR_LASTBRANCH_0_TO_IP.
79H	121	IA32_BIOS_UPDT_TRIG	Unique	BIOS Update Trigger Register (W)
				See Table 2-2.
8BH	139	IA32_BIOS_SIGN_ID	Unique	BIOS Update Signature ID (R/W)
4011	160	MCD CMDD DINGDAGE		See Table 2-2.
AOH	160	MSR_SMRR_PHYSBASE	Unique	System Management Mode Base Address register (WO in SMM)
				Model-specific implementation of SMRR-like interface, read visible and write only in SMM.
		11:0		Reserved
		31:12		PhysBase: SMRR physical Base Address.
		63:32		Reserved

Table 2-3. MSRs in Processors Based on Intel® Core™ Microarchitecture (Contd.)

	ister Iress	Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
A1H	161	MSR_SMRR_PHYSMASK	Unique	System Management Mode Physical Address Mask register (WO in SMM) Model-specific implementation of SMRR-like interface,
				read visible and write only in SMM.
		10:0		Reserved
		11		Valid: Physical address base and range mask are valid.
		31:12		PhysMask: SMRR physical address range mask.
		63:32		Reserved
C1H	193	IA32_PMC0	Unique	Performance Counter Register See Table 2-2.
C2H	194	IA32_PMC1	Unique	Performance Counter Register See Table 2-2.
CDH	205	MSR_FSB_FREQ	Shared	Scaleable Bus Speed (R/O) This field indicates the intended scaleable bus clock speed
				for processors based on Intel Core microarchitecture.
		2:0		 101B: 100 MHz (FSB 400) 001B: 133 MHz (FSB 533) 011B: 167 MHz (FSB 667) 010B: 200 MHz (FSB 800) 000B: 267 MHz (FSB 1067) 100B: 333 MHz (FSB 1333)
				133.33 MHz should be utilized if performing calculation with System Bus Speed when encoding is 001B.
				166.67 MHz should be utilized if performing calculation with System Bus Speed when encoding is 011B.
				266.67 MHz should be utilized if performing calculation with System Bus Speed when encoding is 000B.
				333.33 MHz should be utilized if performing calculation with System Bus Speed when encoding is 100B.
		63:3		Reserved
CDH	205	MSR_FSB_FREQ	Shared	Scaleable Bus Speed (R/O)
				This field indicates the intended scaleable bus clock speed for processors based on Enhanced Intel Core microarchitecture.
		2:0		 101B: 100 MHz (FSB 400) 001B: 133 MHz (FSB 533) 011B: 167 MHz (FSB 667) 010B: 200 MHz (FSB 800) 000B: 267 MHz (FSB 1067) 100B: 333 MHz (FSB 1333) 110B: 400 MHz (FSB 1600)

Table 2-3. MSRs in Processors Based on Intel® Core™ Microarchitecture (Contd.)

Regi Add		Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
				133.33 MHz should be utilized if performing calculation with System Bus Speed when encoding is 001B. 166.67 MHz should be utilized if performing calculation with System Bus Speed when encoding is 011B.
				266.67 MHz should be utilized if performing calculation with System Bus Speed when encoding is 110B. 333.33 MHz should be utilized if performing calculation
				with System Bus Speed when encoding is 111B.
		63:3		Reserved
E7H	231	IA32_MPERF	Unique	Maximum Performance Frequency Clock Count (R/W) See Table 2-2.
E8H	232	IA32_APERF	Unique	Actual Performance Frequency Clock Count (R/W) See Table 2-2.
FEH	254	IA32_MTRRCAP	Unique	See Table 2-2.
		11	Unique	SMRR Capability Using MSR 0A0H and 0A1H (R)
174H	372	IA32_SYSENTER_CS	Unique	See Table 2-2.
175H	373	IA32_SYSENTER_ESP	Unique	See Table 2-2.
176H	374	IA32_SYSENTER_EIP	Unique	See Table 2-2.
179H	377	IA32_MCG_CAP	Unique	See Table 2-2.
17AH	378	IA32_MCG_STATUS	Unique	Global Machine Check Status
		0		RIPV When set, bit indicates that the instruction addressed by the instruction pointer pushed on the stack (when the machine check was generated) can be used to restart the program. If cleared, the program cannot be reliably restarted.
		1		EIPV When set, bit indicates that the instruction addressed by the instruction pointer pushed on the stack (when the machine check was generated) is directly associated with the error.
		2		MCIP When set, bit indicates that a machine check has been generated. If a second machine check is detected while this bit is still set, the processor enters a shutdown state. Software should write this bit to 0 after processing a machine check exception.
		63:3		Reserved
186H	390	IA32_PERFEVTSEL0	Unique	See Table 2-2.
187H	391	IA32_PERFEVTSEL1	Unique	See Table 2-2.
198H	408	IA32_PERF_STATUS	Shared	See Table 2-2.

Table 2-3. MSRs in Processors Based on Intel® Core™ Microarchitecture (Contd.)

Regi Add		Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
198H	408	MSR_PERF_STATUS	Shared	Current performance status. See Section 15.1.1, "Software Interface For Initiating Performance State Transitions".
		15:0		Current Performance State Value
		30:16		Reserved
		31		XE Operation (R/O).
				If set, XE operation is enabled. Default is cleared.
		39:32		Reserved
		44:40		Maximum Bus Ratio (R/O)
				Indicates maximum bus ratio configured for the processor.
		45		Reserved
		46		Non-Integer Bus Ratio (R/O)
				Indicates non-integer bus ratio is enabled. Applies processors based on Enhanced Intel Core microarchitecture.
		63:47		Reserved
199H	409	IA32_PERF_CTL	Unique	See Table 2-2.
19AH	410	IA32_CLOCK_MODULATION	Unique	Clock Modulation (R/W)
				See Table 2-2. IA32_CLOCK_MODULATION MSR was originally named
				IA32_THERM_CONTROL MSR.
19BH	411	IA32_THERM_INTERRUPT	Unique	Thermal Interrupt Control (R/W)
				See Table 2-2.
19CH	412	IA32_THERM_STATUS	Unique	Thermal Monitor Status (R/W)
4.55.1			<u> </u>	See Table 2-2.
19DH	413	MSR_THERM2_CTL	Unique	Thermal Monitor 2 Control
		15:0		Reserved
		16		TM_SELECT (R/W)
				Mode of automatic thermal monitor:
				0 = Thermal Monitor 1 (thermally-initiated on-die modulation of the stop-clock duty cycle).
				1 = Thermal Monitor 2 (thermally-initiated frequency transitions).
				If bit 3 of the IA32_MISC_ENABLE register is cleared, TM_SELECT has no effect. Neither TM1 nor TM2 are enabled.
		63:16		Reserved
1A0H	416	IA32_MISC_ENABLE		Enable Misc. Processor Features (R/W)
				Allows a variety of processor functions to be enabled and disabled.

Table 2-3. MSRs in Processors Based on Intel® Core™ Microarchitecture (Contd.)

Regi Add		Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
		0		Fast-Strings Enable See Table 2-2.
		2:1		Reserved
		3	Unique	Automatic Thermal Control Circuit Enable (R/W) See Table 2-2.
		6:4		Reserved
		7	Shared	Performance Monitoring Available (R) See Table 2-2.
		8		Reserved
		9		Hardware Prefetcher Disable (R/W) When set, disables the hardware prefetcher operation on streams of data. When clear (default), enables the prefetch queue. Disabling of the hardware prefetcher may impact processor performance.
		10	Shared	FERR# Multiplexing Enable (R/W) 1 = FERR# asserted by the processor to indicate a pending break event within the processor. 0 = Indicates compatible FERR# signaling behavior. This bit must be set to 1 to support XAPIC interrupt model usage.
		11	Shared	Branch Trace Storage Unavailable (R/O) See Table 2-2.
		12	Shared	Processor Event Based Sampling Unavailable (R/O) See Table 2-2.
		13	Shared	TM2 Enable (R/W) When this bit is set (1) and the thermal sensor indicates that the die temperature is at the pre-determined threshold, the Thermal Monitor 2 mechanism is engaged. TM2 will reduce the bus to core ratio and voltage according to the value last written to MSR_THERM2_CTL bits 15:0.
				When this bit is clear (0, default), the processor does not change the VID signals or the bus to core ratio when the processor enters a thermally managed state.
				The BIOS must enable this feature if the TM2 feature flag (CPUID.1:ECX[8]) is set; if the TM2 feature flag is not set, this feature is not supported and BIOS must not alter the contents of the TM2 bit location.
				The processor is operating out of specification if both this bit and the TM1 bit are set to 0.
		15:14		Reserved

Table 2-3. MSRs in Processors Based on Intel® Core™ Microarchitecture (Contd.)

Regi Addı		Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
		16	Shared	Enhanced Intel SpeedStep Technology Enable (R/W) See Table 2-2.
		18	Shared	ENABLE MONITOR FSM (R/W) See Table 2-2.
		19	Shared	Adjacent Cache Line Prefetch Disable (R/W)
				When set to 1, the processor fetches the cache line that contains data currently required by the processor. When set to 0, the processor fetches cache lines that comprise a cache line pair (128 bytes).
				Single processor platforms should not set this bit. Server platforms should set or clear this bit based on platform performance observed in validation and testing.
				BIOS may contain a setup option that controls the setting of this bit.
		20	Shared	Enhanced Intel SpeedStep Technology Select Lock (R/WO) When set, this bit causes the following bits to become read-only:
				 Enhanced Intel SpeedStep Technology Select Lock (this bit). Enhanced Intel SpeedStep Technology Enable bit.
				The bit must be set before an Enhanced Intel SpeedStep Technology transition is requested. This bit is cleared on reset.
		21		Reserved
		22	Shared	Limit CPUID Maxval (R/W) See Table 2-2.
		23	Shared	xTPR Message Disable (R/W) See Table 2-2.
		33:24		Reserved
		34	Unique	XD Bit Disable (R/W) See Table 2-2.
		36:35		Reserved
		37	Unique	DCU Prefetcher Disable (R/W)
				When set to 1, the DCU L1 data cache prefetcher is disabled. The default value after reset is 0. BIOS may write '1' to disable this feature.
				The DCU prefetcher is an L1 data cache prefetcher. When the DCU prefetcher detects multiple loads from the same line done within a time limit, the DCU prefetcher assumes the next line will be required. The next line is prefetched in to the L1 data cache from memory or L2.

Table 2-3. MSRs in Processors Based on Intel® Core™ Microarchitecture (Contd.)

Regi Add	ster ress	Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
		38	Shared	IDA Disable (R/W) When set to 1 on processors that support IDA, the Intel Dynamic Acceleration feature (IDA) is disabled and the IDA_Enable feature flag will be cleared (CPUID.06H: EAX[1]=0).
				When set to a 0 on processors that support IDA, CPUID.06H: EAX[1] reports the processor's support of IDA is enabled.
				Note: The power-on default value is used by BIOS to detect hardware support of IDA. If the power-on default value is 1, IDA is available in the processor. If the power-on default value is 0, IDA is not available.
		39	Unique	IP Prefetcher Disable (R/W)
				When set to 1, the IP prefetcher is disabled. The default value after reset is 0. BIOS may write '1' to disable this feature.
				The IP prefetcher is an L1 data cache prefetcher. The IP prefetcher looks for sequential load history to determine whether to prefetch the next expected data into the L1 cache from memory or L2.
		63:40		Reserved
1C9H	457	MSR_LASTBRANCH_TOS	Unique	Last Branch Record Stack TOS (R/W)
				Contains an index (bits 0-3) that points to the MSR containing the most recent branch record.
				See MSR_LASTBRANCH_0_FROM_IP (at 40H).
1D9H	473	IA32_DEBUGCTL	Unique	Debug Control (R/W) See Table 2-2.
1DDH	477	MSR_LER_FROM_LIP	Unique	Last Exception Record From Linear IP (R/W)
				Contains a pointer to the last branch instruction that the processor executed prior to the last exception that was generated or the last interrupt that was handled.
1DEH	478	MSR_LER_TO_LIP	Unique	Last Exception Record To Linear IP (R/W)
				This area contains a pointer to the target of the last branch instruction that the processor executed prior to the last exception that was generated or the last interrupt that was handled.
200H	512	IA32_MTRR_PHYSBASE0	Unique	See Table 2-2.
201H	513	IA32_MTRR_PHYSMASKO	Unique	See Table 2-2.
202H	514	IA32_MTRR_PHYSBASE1	Unique	See Table 2-2.
203H	515	IA32_MTRR_PHYSMASK1	Unique	See Table 2-2.
204H	516	IA32_MTRR_PHYSBASE2	Unique	See Table 2-2.
205H	517	IA32_MTRR_PHYSMASK2	Unique	See Table 2-2.

Table 2-3. MSRs in Processors Based on Intel® Core™ Microarchitecture (Contd.)

Regi Add	ister ress	Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
206H	518	IA32_MTRR_PHYSBASE3	Unique	See Table 2-2.
207H	519	IA32_MTRR_PHYSMASK3	Unique	See Table 2-2.
208H	520	IA32_MTRR_PHYSBASE4	Unique	See Table 2-2.
209H	521	IA32_MTRR_PHYSMASK4	Unique	See Table 2-2.
20AH	522	IA32_MTRR_PHYSBASE5	Unique	See Table 2-2.
20BH	523	IA32_MTRR_PHYSMASK5	Unique	See Table 2-2.
20CH	524	IA32_MTRR_PHYSBASE6	Unique	See Table 2-2.
20DH	525	IA32_MTRR_PHYSMASK6	Unique	See Table 2-2.
20EH	526	IA32_MTRR_PHYSBASE7	Unique	See Table 2-2.
20FH	527	IA32_MTRR_PHYSMASK7	Unique	See Table 2-2.
250H	592	IA32_MTRR_FIX64K_00000	Unique	See Table 2-2.
258H	600	IA32_MTRR_FIX16K_80000	Unique	See Table 2-2.
259H	601	IA32_MTRR_FIX16K_A0000	Unique	See Table 2-2.
268H	616	IA32_MTRR_FIX4K_C0000	Unique	See Table 2-2.
269H	617	IA32_MTRR_FIX4K_C8000	Unique	See Table 2-2.
26AH	618	IA32_MTRR_FIX4K_D0000	Unique	See Table 2-2.
26BH	619	IA32_MTRR_FIX4K_D8000	Unique	See Table 2-2.
26CH	620	IA32_MTRR_FIX4K_E0000	Unique	See Table 2-2.
26DH	621	IA32_MTRR_FIX4K_E8000	Unique	See Table 2-2.
26EH	622	IA32_MTRR_FIX4K_F0000	Unique	See Table 2-2.
26FH	623	IA32_MTRR_FIX4K_F8000	Unique	See Table 2-2.
277H	631	IA32_PAT	Unique	See Table 2-2.
2FFH	767	IA32_MTRR_DEF_TYPE	Unique	Default Memory Types (R/W)
				See Table 2-2.
309H	777	IA32_FIXED_CTR0	Unique	Fixed-Function Performance Counter Register 0 (R/W) See Table 2-2.
30AH	778	IA32_FIXED_CTR1	Unique	Fixed-Function Performance Counter Register 1 (R/W) See Table 2-2.
30BH	779	IA32_FIXED_CTR2	Unique	Fixed-Function Performance Counter Register 2 (R/W) See Table 2-2.
345H	837	IA32_PERF_CAPABILITIES	Unique	See Table 2-2. See Section 18.4.1, "IA32_DEBUGCTL MSR."
345H	837	MSR_PERF_CAPABILITIES	Unique	R/O. This applies to processors that do not support architectural perfmon version 2.

Table 2-3. MSRs in Processors Based on Intel® Core™ Microarchitecture (Contd.)

	ister ress	Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
		5:0		LBR Format. See Table 2-2.
		6		PEBS Record Format
		7		PEBSSaveArchRegs. See Table 2-2.
		63:8		Reserved
38DH	909	IA32_FIXED_CTR_CTRL	Unique	Fixed-Function-Counter Control Register (R/W)
				See Table 2-2.
38EH	910	IA32_PERF_GLOBAL_STATUS	Unique	See Table 2-2. See Section 20.6.2.2, "Global Counter Control Facilities."
38EH	910	MSR_PERF_GLOBAL_STATUS	Unique	See Section 20.6.2.2, "Global Counter Control Facilities."
38FH	911	IA32_PERF_GLOBAL_CTRL	Unique	See Table 2-2. See Section 20.6.2.2, "Global Counter Control Facilities."
38FH	911	MSR_PERF_GLOBAL_CTRL	Unique	See Section 20.6.2.2, "Global Counter Control Facilities."
390H	912	IA32_PERF_GLOBAL_OVF_CTRL	Unique	See Table 2-2. See Section 20.6.2.2, "Global Counter Control Facilities."
390H	912	MSR_PERF_GLOBAL_OVF_CTRL	Unique	See Section 20.6.2.2, "Global Counter Control Facilities."
3F1H	1009	IA32_PEBS_ENABLE (MSR_PEBS_ENABLE)	Unique	See Table 2-2. See Section 20.6.2.4, "Processor Event Based Sampling (PEBS)."
		0		Enable PEBS on IA32_PMCO. (R/W)
400H	1024	IA32_MCO_CTL	Unique	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
401H	1025	IA32_MCO_STATUS	Unique	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
402H	1026	IA32_MCO_ADDR	Unique	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
				The IA32_MC0_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MC0_STATUS register is clear.
				When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
404H	1028	IA32_MC1_CTL	Unique	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
405H	1029	IA32_MC1_STATUS	Unique	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
406H	1030	IA32_MC1_ADDR	Unique	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
				The IA32_MC1_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MC1_STATUS register is clear.
				When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
408H	1032	IA32_MC2_CTL	Unique	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
409H	1033	IA32_MC2_STATUS	Unique	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."

Table 2-3. MSRs in Processors Based on Intel® Core™ Microarchitecture (Contd.)

	ister ress	Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
40AH	1034	IA32_MC2_ADDR	Unique	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs." The IA32_MC2_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MC2_STATUS register is clear.
				When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
40CH	1036	IA32_MC4_CTL	Unique	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
40DH	1037	IA32_MC4_STATUS	Unique	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
40EH	1038	IA32_MC4_ADDR	Unique	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
				The MSR_MC4_ADDR register is either not implemented or contains no address if the ADDRV flag in the MSR_MC4_STATUS register is clear.
				When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
410H	1040	IA32_MC3_CTL		See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
411H	1041	IA32_MC3_STATUS		See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
412H	1042	IA32_MC3_ADDR	Unique	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
				The MSR_MC3_ADDR register is either not implemented or contains no address if the ADDRV flag in the MSR_MC3_STATUS register is clear.
				When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
413H	1043	IA32_MC3_MISC	Unique	Machine Check Error Reporting Register: Contains additional information describing the machine-check error if the MISCV flag in the IA32_MCi_STATUS register is set.
414H	1044	IA32_MC5_CTL	Unique	Machine Check Error Reporting Register: Controls signaling of #MC for errors produced by a particular hardware unit (or group of hardware units).
415H	1045	IA32_MC5_STATUS	Unique	Machine Check Error Reporting Register: Contains information related to a machine-check error if its VAL (valid) flag is set. Software is responsible for clearing IA32_MCi_STATUS MSRs by explicitly writing 0s to them; writing 1s to them causes a general-protection exception.
416H	1046	IA32_MC5_ADDR	Unique	Machine Check Error Reporting Register: Contains the address of the code or data memory location that produced the machine-check error if the ADDRV flag in the IA32_MCi_STATUS register is set.
417H	1047	IA32_MC5_MISC	Unique	Machine Check Error Reporting Register: Contains additional information describing the machine-check error if the MISCV flag in the IA32_MCi_STATUS register is set.

Table 2-3. MSRs in Processors Based on Intel® Core™ Microarchitecture (Contd.)

	ister ress	Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
419H	1045	IA32_MC6_STATUS	Unique	Applies to Intel Xeon processor 7400 series (processor signature 06_1D) only. See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 24.
480H	1152	IA32_VMX_BASIC	Unique	Reporting Register of Basic VMX Capabilities (R/O) See Table 2-2. See Appendix A.1, "Basic VMX Information."
481H	1153	IA32_VMX_PINBASED_CTLS	Unique	Capability Reporting Register of Pin-Based VM-Execution Controls (R/O) See Table 2-2. See Appendix A.3, "VM-Execution Controls."
482H	1154	IA32_VMX_PROCBASED_CTLS	Unique	Capability Reporting Register of Primary Processor-Based VM-Execution Controls (R/O) See Appendix A.3, "VM-Execution Controls."
483H	1155	IA32_VMX_EXIT_CTLS	Unique	Capability Reporting Register of VM-Exit Controls (R/O) See Table 2-2. See Appendix A.4, "VM-Exit Controls."
484H	1156	IA32_VMX_ENTRY_CTLS	Unique	Capability Reporting Register of VM-Entry Controls (R/O) See Table 2-2. See Appendix A.5, "VM-Entry Controls."
485H	1157	IA32_VMX_MISC	Unique	Reporting Register of Miscellaneous VMX Capabilities (R/O) See Table 2-2. See Appendix A.6, "Miscellaneous Data."
486H	1158	IA32_VMX_CR0_FIXED0	Unique	Capability Reporting Register of CRO Bits Fixed to 0 (R/O) See Table 2-2. See Appendix A.7, "VMX-Fixed Bits in CRO."
487H	1159	IA32_VMX_CRO_FIXED1	Unique	Capability Reporting Register of CRO Bits Fixed to 1 (R/O) See Table 2-2. See Appendix A.7, "VMX-Fixed Bits in CRO."
488H	1160	IA32_VMX_CR4_FIXED0	Unique	Capability Reporting Register of CR4 Bits Fixed to 0 (R/0) See Table 2-2. See Appendix A.8, "VMX-Fixed Bits in CR4."
489H	1161	IA32_VMX_CR4_FIXED1	Unique	Capability Reporting Register of CR4 Bits Fixed to 1 (R/0) See Table 2-2. See Appendix A.8, "VMX-Fixed Bits in CR4."
48AH	1162	IA32_VMX_VMCS_ENUM	Unique	Capability Reporting Register of VMCS Field Enumeration (R/O) See Table 2-2. See Appendix A.9, "VMCS Enumeration."

Table 2-3. MSRs in Processors Based on Intel® Core™ Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
48BH	1163	IA32_VMX_PROCBASED_CTLS2	Unique	Capability Reporting Register of Secondary Processor- Based VM-Execution Controls (R/O)
				See Appendix A.3, "VM-Execution Controls."
600H	1536	IA32_DS_AREA	Unique	DS Save Area (R/W)
				See Table 2-2.
				See Section 20.6.3.4, "Debug Store (DS) Mechanism."
107CC	67532	MSR_EMON_L3_CTR_CTL0	Unique	GBUSQ Event Control/Counter Register (R/W)
Н				Applies to Intel Xeon processor 7400 series (processor signature 06_1D) only. See Section 18.2.2
107CD	67533	MSR_EMON_L3_CTR_CTL1	Unique	GBUSQ Event Control/Counter Register (R/W)
H				Applies to Intel Xeon processor 7400 series (processor signature 06_1D) only. See Section 18.2.2
107CE	67534	MSR_EMON_L3_CTR_CTL2	Unique	GSNPQ Event Control/Counter Register (R/W)
H				Applies to Intel Xeon processor 7400 series (processor signature 06_1D) only. See Section 18.2.2
107CF	67535	MSR_EMON_L3_CTR_CTL3	Unique	GSNPQ Event Control/Counter Register (R/W)
Н				Applies to Intel Xeon processor 7400 series (processor signature 06_1D) only. See Section 18.2.2
107D0	67536	MSR_EMON_L3_CTR_CTL4	Unique	FSB Event Control/Counter Register (R/W)
Н				Applies to Intel Xeon processor 7400 series (processor signature 06_1D) only. See Section 18.2.2
107D1	67537	MSR_EMON_L3_CTR_CTL5	Unique	FSB Event Control/Counter Register (R/W)
Н				Applies to Intel Xeon processor 7400 series (processor signature 06_1D) only. See Section 18.2.2
107D2	67538	MSR_EMON_L3_CTR_CTL6	Unique	FSB Event Control/Counter Register (R/W)
Н				Applies to Intel Xeon processor 7400 series (processor signature 06_1D) only. See Section 18.2.2
107D3	67539	MSR_EMON_L3_CTR_CTL7	Unique	FSB Event Control/Counter Register (R/W)
Н				Applies to Intel Xeon processor 7400 series (processor signature 06_1D) only. See Section 18.2.2
107D8	67544	MSR_EMON_L3_GL_CTL	Unique	L3/FSB Common Control Register (R/W)
Н				Applies to Intel Xeon processor 7400 series (processor signature 06_1D) only. See Section 18.2.2
C000_		IA32_EFER	Unique	Extended Feature Enables
0080H				See Table 2-2.
C000_		IA32_STAR	Unique	System Call Target Address (R/W)
0081H				See Table 2-2.
C000_		IA32_LSTAR	Unique	IA-32e Mode System Call Target Address (R/W)
0082H				See Table 2-2.

Table 2-3. MSRs in Processors Based on Intel® Core™ Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
C000_ 0084H		IA32_FMASK	Unique	System Call Flag Mask (R/W) See Table 2-2.
C000_ 0100H		IA32_FS_BASE	Unique	Map of BASE Address of FS (R/W) See Table 2-2.
C000_ 0101H		IA32_GS_BASE	Unique	Map of BASE Address of GS (R/W) See Table 2-2.
C000_ 0102H		IA32_KERNEL_GS_BASE	Unique	Swap Target of BASE Address of GS (R/W) See Table 2-2.

2.3 MSRS IN THE 45 NM AND 32 NM INTEL ATOM® PROCESSOR FAMILY

Table 2-4 lists model-specific registers (MSRs) for 45 nm and 32 nm Intel Atom processors, architectural MSR addresses are also included in Table 2-4. These processors have a CPUID signature with DisplayFamily_DisplayModel of 06_1CH, 06_26H, 06_27H, 06_35H, and 06_36H; see Table 2-1.

The column "Shared/Unique" applies to logical processors sharing the same core in processors based on the Intel Atom microarchitecture. "Unique" means each logical processor has a separate MSR, or a bit field in an MSR governs only a logical processor. "Shared" means the MSR or the bit field in an MSR address governs the operation of both logical processors in the same core.

Table 2-4. MSRs in 45 nm and 32 nm Intel Atom® Processor Family

Register Address		Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
OH	0	IA32_P5_MC_ADDR	Shared	See Section 2.23, "MSRs in Pentium Processors."
1H	1	IA32_P5_MC_TYPE	Shared	See Section 2.23, "MSRs in Pentium Processors."
6H	6	IA32_MONITOR_FILTER_ SIZE	Unique	See Section 9.10.5, "Monitor/Mwait Address Range Determination." and Table 2-2.
10H	16	IA32_TIME_STAMP_ COUNTER	Unique	See Section 18.17, "Time-Stamp Counter," and see Table 2-2.
17H	23	IA32_PLATFORM_ID	Shared	Platform ID (R) See Table 2-2.
17H	23	MSR_PLATFORM_ID	Shared	Model Specific Platform ID (R)
		7:0		Reserved
		12:8		Maximum Qualified Ratio (R)
				The maximum allowed bus ratio.
		63:13		Reserved
1BH	27	IA32_APIC_BASE	Unique	See Section 11.4.4, "Local APIC Status and Location" and Table 2-2.

Table 2-4. MSRs in 45 nm and 32 nm Intel Atom® Processor Family (Contd.)

Regi Add	ister ress	Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
2AH	42	MSR_EBL_CR_POWERON	Shared	Processor Hard Power-On Configuration (R/W) Enables and disables processor features; (R) indicates current processor configuration.
		0		Reserved
		1		Data Error Checking Enable (R/W) 1 = Enabled; 0 = Disabled Always 0.
		2		Response Error Checking Enable (R/W) 1 = Enabled; 0 = Disabled Always 0.
		3		AERR# Drive Enable (R/W) 1 = Enabled; 0 = Disabled Always 0.
		4		BERR# Enable for initiator bus requests (R/W) 1 = Enabled; 0 = Disabled Always 0.
		5		Reserved
		6		Reserved
		7		BINIT# Driver Enable (R/W) 1 = Enabled; 0 = Disabled Always 0.
		8		Reserved
		9		Execute BIST (R/O) 1 = Enabled; 0 = Disabled
		10		AERR# Observation Enabled (R/O) 1 = Enabled; O = Disabled Always O.
		11		Reserved
		12		BINIT# Observation Enabled (R/O) 1 = Enabled; 0 = Disabled Always 0.
		13		Reserved
		14		1 MByte Power on Reset Vector (R/O) 1 = 1 MByte; 0 = 4 GBytes
		15		Reserved
		17:16		APIC Cluster ID (R/O) Always 00B.
		19: 18		Reserved
		21: 20		Symmetric Arbitration ID (R/O) Always 00B.

Table 2-4. MSRs in 45 nm and 32 nm Intel Atom® Processor Family (Contd.)

	ister ress	Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
		26:22		Integer Bus Frequency Ratio (R/O)
ЗАН	58	IA32_FEATURE_CONTROL	Unique	Control Features in Intel 64Processor (R/W) See Table 2-2.
40H	64	MSR_LASTBRANCH_0_FROM_IP	Unique	Last Branch Record 0 From IP (R/W) One of eight pairs of last branch record registers on the last branch record stack. The From_IP part of the stack contains pointers to the source instruction . See also: Last Branch Record Stack TOS at 1C9H. Section 18.5.
41H	65	MSR_LASTBRANCH_1_FROM_IP	Unique	Last Branch Record 1 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
42H	66	MSR_LASTBRANCH_2_FROM_IP	Unique	Last Branch Record 2 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
43H	67	MSR_LASTBRANCH_3_FROM_IP	Unique	Last Branch Record 3 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
44H	68	MSR_LASTBRANCH_4_FROM_IP	Unique	Last Branch Record 4 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
45H	69	MSR_LASTBRANCH_5_FROM_IP	Unique	Last Branch Record 5 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
46H	70	MSR_LASTBRANCH_6_FROM_IP	Unique	Last Branch Record 6 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
47H	71	MSR_LASTBRANCH_7_FROM_IP	Unique	Last Branch Record 7 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
60H	96	MSR_LASTBRANCH_0_TO_IP	Unique	Last Branch Record 0 To IP (R/W) One of eight pairs of last branch record registers on the last branch record stack. The To_IP part of the stack contains pointers to the destination instruction.
61H	97	MSR_LASTBRANCH_1_TO_IP	Unique	Last Branch Record 1 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
62H	98	MSR_LASTBRANCH_2_TO_IP	Unique	Last Branch Record 2 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
63H	99	MSR_LASTBRANCH_3_TO_IP	Unique	Last Branch Record 3 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
64H	100	MSR_LASTBRANCH_4_TO_IP	Unique	Last Branch Record 4 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
65H	101	MSR_LASTBRANCH_5_TO_IP	Unique	Last Branch Record 5 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
66H	102	MSR_LASTBRANCH_6_TO_IP	Unique	Last Branch Record 6 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
67H	103	MSR_LASTBRANCH_7_TO_IP	Unique	Last Branch Record 7 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.

Table 2-4. MSRs in 45 nm and 32 nm Intel Atom® Processor Family (Contd.)

	ister ress	Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
79H	121	IA32_BIOS_UPDT_TRIG	Shared	BIOS Update Trigger Register (W) See Table 2-2.
8BH	139	IA32_BIOS_SIGN_ID	Unique	BIOS Update Signature ID (R/W) See Table 2-2.
C1H	193	IA32_PMC0	Unique	Performance counter register See Table 2-2.
C2H	194	IA32_PMC1	Unique	Performance Counter Register See Table 2-2.
CDH	205	MSR_FSB_FREQ	Shared	Scaleable Bus Speed (R/O) This field indicates the intended scaleable bus clock speed for processors based on Intel Atom microarchitecture.
		2:0		 111B: 083 MHz (FSB 333) 101B: 100 MHz (FSB 400) 001B: 133 MHz (FSB 533) 011B: 167 MHz (FSB 667)
				133.33 MHz should be utilized if performing calculation with System Bus Speed when encoding is 001B. 166.67 MHz should be utilized if performing calculation with System Bus Speed when encoding is 011B.
		63:3		Reserved
E7H	231	IA32_MPERF	Unique	Maximum Performance Frequency Clock Count (R/W) See Table 2-2.
E8H	232	IA32_APERF	Unique	Actual Performance Frequency Clock Count (R/W) See Table 2-2.
FEH	254	IA32_MTRRCAP	Shared	Memory Type Range Register (R) See Table 2-2.
11EH	281	MSR_BBL_CR_CTL3	Shared	Control Register 3 Used to configure the L2 Cache.
		0		L2 Hardware Enabled (R/O) 1 = Indicates the L2 is hardware-enabled. 0 = Indicates the L2 is hardware-disabled.
		7:1		Reserved
		8		L2 Enabled (R/W) 1 = L2 cache has been initialized. 0 = Disabled (default). Until this bit is set, the processor will not respond to the WBINVD instruction or the assertion of the FLUSH# input.
		22:9		Reserved
		23 63:24		L2 Not Present (R/O) 0 = L2 Present 1 = L2 Not Present Reserved
		UJ.LT		I/C2CI AGA

Table 2-4. MSRs in 45 nm and 32 nm Intel Atom® Processor Family (Contd.)

Regi Add		Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
174H	372	IA32_SYSENTER_CS	Unique	See Table 2-2.
175H	373	IA32_SYSENTER_ESP	Unique	See Table 2-2.
176H	374	IA32_SYSENTER_EIP	Unique	See Table 2-2.
179H	377	IA32_MCG_CAP	Unique	See Table 2-2.
17AH	378	IA32_MCG_STATUS	Unique	Global Machine Check Status
		0		RIPV
				When set, bit indicates that the instruction addressed by the instruction pointer pushed on the stack (when the machine check was generated) can be used to restart the program. If cleared, the program cannot be reliably restarted.
		1		EIPV
				When set, bit indicates that the instruction addressed by the instruction pointer pushed on the stack (when the machine check was generated) is directly associated with the error.
		2		MCIP
				When set, bit indicates that a machine check has been generated. If a second machine check is detected while this bit is still set, the processor enters a shutdown state. Software should write this bit to 0 after processing a machine check exception.
		63:3		Reserved
186H	390	IA32_PERFEVTSEL0	Unique	See Table 2-2.
187H	391	IA32_PERFEVTSEL1	Unique	See Table 2-2.
198H	408	IA32_PERF_STATUS	Shared	See Table 2-2.
198H	408	MSR_PERF_STATUS	Shared	Performance Status
		15:0		Current Performance State Value
		39:16		Reserved
		44:40		Maximum Bus Ratio (R/O)
				Indicates maximum bus ratio configured for the processor.
		63:45		Reserved
199H	409	IA32_PERF_CTL	Unique	See Table 2-2.
19AH	410	IA32_CLOCK_MODULATION	Unique	Clock Modulation (R/W)
				See Table 2-2. IA32_CLOCK_MODULATION MSR was originally named IA32_THERM_CONTROL MSR.
19BH	411	IA32_THERM_INTERRUPT	Unique	Thermal Interrupt Control (R/W) See Table 2-2.
19CH	412	IA32_THERM_STATUS	Unique	Thermal Monitor Status (R/W) See Table 2-2.
19DH	413	MSR_THERM2_CTL	Shared	Thermal Monitor 2 Control

Table 2-4. MSRs in 45 nm and 32 nm Intel Atom® Processor Family (Contd.)

	ister ress	Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
		15:0		Reserved
		16		 TM_SELECT (R/W) Mode of automatic thermal monitor: 0 = Thermal Monitor 1 (thermally-initiated on-die modulation of the stop-clock duty cycle). 1 = Thermal Monitor 2 (thermally-initiated frequency transitions). If bit 3 of the IA32_MISC_ENABLE register is cleared,
				TM_SELECT has no effect. Neither TM1 nor TM2 are enabled.
		63:17		Reserved
1A0H	416	IA32_MISC_ENABLE	Unique	Enable Misc. Processor Features (R/W) Allows a variety of processor functions to be enabled and disabled.
		0		Fast-Strings Enable See Table 2-2.
		2:1		Reserved
		3	Unique	Automatic Thermal Control Circuit Enable (R/W) See Table 2-2. Default value is 0.
		6:4		Reserved
		7	Shared	Performance Monitoring Available (R) See Table 2-2.
		8		Reserved
		9		Reserved
		10	Shared	FERR# Multiplexing Enable (R/W) 1 = FERR# asserted by the processor to indicate a pending break event within the processor. 0 = Indicates compatible FERR# signaling behavior. This bit must be set to 1 to support XAPIC interrupt model usage.
		11	Shared	Branch Trace Storage Unavailable (R/O) See Table 2-2.
		12	Shared	Processor Event Based Sampling Unavailable (R/O) See Table 2-2.
		13	Shared	TM2 Enable (R/W) When this bit is set (1) and the thermal sensor indicates that the die temperature is at the pre-determined threshold, the Thermal Monitor 2 mechanism is engaged. TM2 will reduce the bus to core ratio and voltage according to the value last written to MSR_THERM2_CTL bits 15:0.

Table 2-4. MSRs in 45 nm and 32 nm Intel Atom® Processor Family (Contd.)

Regi Add		Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
				When this bit is cleared (0, default), the processor does not change the VID signals or the bus to core ratio when the processor enters a thermally managed state.
				The BIOS must enable this feature if the TM2 feature flag (CPUID.1:ECX[8]) is set; if the TM2 feature flag is not set, this feature is not supported and BIOS must not alter the contents of the TM2 bit location.
				The processor is operating out of specification if both this bit and the TM1 bit are set to 0.
		15:14		Reserved
		16	Shared	Enhanced Intel SpeedStep Technology Enable (R/W) See Table 2-2.
		18	Shared	ENABLE MONITOR FSM (R/W)
				See Table 2-2.
		19		Reserved
		20	Shared	Enhanced Intel SpeedStep Technology Select Lock (R/WO) When set, this bit causes the following bits to become read-only:
				 Enhanced Intel SpeedStep Technology Select Lock (this bit). Enhanced Intel SpeedStep Technology Enable bit.
				The bit must be set before an Enhanced Intel SpeedStep Technology transition is requested. This bit is cleared on reset.
		21		Reserved
		22	Unique	Limit CPUID Maxval (R/W) See Table 2-2.
		23	Shared	xTPR Message Disable (R/W) See Table 2-2.
		33:24		Reserved
		34	Unique	XD Bit Disable (R/W)
				See Table 2-2.
		63:35		Reserved
1C9H	457	MSR_LASTBRANCH_TOS	Unique	Last Branch Record Stack TOS (R/W)
				Contains an index (bits 0-2) that points to the MSR containing the most recent branch record.
				See MSR_LASTBRANCH_0_FROM_IP (at 40H).
1D9H	473	IA32_DEBUGCTL	Unique	Debug Control (R/W)
10011	477	MCD LCD CDOM LID	Helevis	See Table 2-2.
1DDH	477	MSR_LER_FROM_LIP	Unique	Last Exception Record From Linear IP (R) Contains a pointer to the last branch instruction that the processor executed prior to the last exception that was generated or the last interrupt that was handled.

Table 2-4. MSRs in 45 nm and 32 nm Intel Atom® Processor Family (Contd.)

Register Address		Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
1DEH	478	MSR_LER_TO_LIP	Unique	Last Exception Record To Linear IP (R)
				This area contains a pointer to the target of the last branch instruction that the processor executed prior to the last exception that was generated or the last interrupt that was handled.
200H	512	IA32_MTRR_PHYSBASE0	Shared	See Table 2-2.
201H	513	IA32_MTRR_PHYSMASK0	Shared	See Table 2-2.
202H	514	IA32_MTRR_PHYSBASE1	Shared	See Table 2-2.
203H	515	IA32_MTRR_PHYSMASK1	Shared	See Table 2-2.
204H	516	IA32_MTRR_PHYSBASE2	Shared	See Table 2-2.
205H	517	IA32_MTRR_PHYSMASK2	Shared	See Table 2-2.
206H	518	IA32_MTRR_PHYSBASE3	Shared	See Table 2-2.
207H	519	IA32_MTRR_PHYSMASK3	Shared	See Table 2-2.
208H	520	IA32_MTRR_PHYSBASE4	Shared	See Table 2-2.
209H	521	IA32_MTRR_PHYSMASK4	Shared	See Table 2-2.
20AH	522	IA32_MTRR_PHYSBASE5	Shared	See Table 2-2.
20BH	523	IA32_MTRR_PHYSMASK5	Shared	See Table 2-2.
20CH	524	IA32_MTRR_PHYSBASE6	Shared	See Table 2-2.
20DH	525	IA32_MTRR_PHYSMASK6	Shared	See Table 2-2.
20EH	526	IA32_MTRR_PHYSBASE7	Shared	See Table 2-2.
20FH	527	IA32_MTRR_PHYSMASK7	Shared	See Table 2-2.
250H	592	IA32_MTRR_FIX64K_00000	Shared	See Table 2-2.
258H	600	IA32_MTRR_FIX16K_80000	Shared	See Table 2-2.
259H	601	IA32_MTRR_FIX16K_A0000	Shared	See Table 2-2.
268H	616	IA32_MTRR_FIX4K_C0000	Shared	See Table 2-2.
269H	617	IA32_MTRR_FIX4K_C8000	Shared	See Table 2-2.
26AH	618	IA32_MTRR_FIX4K_D0000	Shared	See Table 2-2.
26BH	619	IA32_MTRR_FIX4K_D8000	Shared	See Table 2-2.
26CH	620	IA32_MTRR_FIX4K_E0000	Shared	See Table 2-2.
26DH	621	IA32_MTRR_FIX4K_E8000	Shared	See Table 2-2.
26EH	622	IA32_MTRR_FIX4K_F0000	Shared	See Table 2-2.
26FH	623	IA32_MTRR_FIX4K_F8000	Shared	See Table 2-2.
277H	631	IA32_PAT	Unique	See Table 2-2.
309H	777	IA32_FIXED_CTR0	Unique	Fixed-Function Performance Counter Register 0 (R/W) See Table 2-2.
30AH	778	IA32_FIXED_CTR1	Unique	Fixed-Function Performance Counter Register 1 (R/W) See Table 2-2.

Table 2-4. MSRs in 45 nm and 32 nm Intel Atom® Processor Family (Contd.)

	ister ress	Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
30BH	779	IA32_FIXED_CTR2	Unique	Fixed-Function Performance Counter Register 2 (R/W) See Table 2-2.
345H	837	IA32_PERF_CAPABILITIES	Shared	See Table 2-2. See Section 18.4.1, "IA32_DEBUGCTL MSR."
38DH	909	IA32_FIXED_CTR_CTRL	Unique	Fixed-Function-Counter Control Register (R/W) See Table 2-2.
38EH	910	IA32_PERF_GLOBAL_STATUS	Unique	See Table 2-2. See Section 20.6.2.2, "Global Counter Control Facilities."
38FH	911	IA32_PERF_GLOBAL_CTRL	Unique	See Table 2-2. See Section 20.6.2.2, "Global Counter Control Facilities."
390H	912	IA32_PERF_GLOBAL_OVF_CTRL	Unique	See Table 2-2. See Section 20.6.2.2, "Global Counter Control Facilities."
3F1H	1009	IA32_PEBS_ENABLE (MSR_PEBS_ENABLE)	Unique	See Table 2-2. See Section 20.6.2.4, "Processor Event Based Sampling (PEBS)."
		0		Enable PEBS on IA32_PMC0 (R/W)
400H	1024	IA32_MCO_CTL	Shared	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
401H	1025	IA32_MCO_STATUS	Shared	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
402H	1026	IA32_MCO_ADDR	Shared	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs." The IA32_MCO_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MCO_STATUS register is clear.
				When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
404H	1028	IA32_MC1_CTL	Shared	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
405H	1029	IA32_MC1_STATUS	Shared	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
408H	1032	IA32_MC2_CTL	Shared	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
409H	1033	IA32_MC2_STATUS	Shared	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
40AH	1034	IA32_MC2_ADDR	Shared	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
				The IA32_MC2_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MC2_STATUS register is clear.
				When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
40CH	1036	IA32_MC3_CTL	Shared	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
40DH	1037	IA32_MC3_STATUS	Shared	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
40EH	1038	IA32_MC3_ADDR	Shared	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
				The MSR_MC3_ADDR register is either not implemented or contains no address if the ADDRV flag in the MSR_MC3_STATUS register is clear.
				When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.

Table 2-4. MSRs in 45 nm and 32 nm Intel Atom® Processor Family (Contd.)

Regi Add	ister ress	Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
410H	1040	IA32_MC4_CTL	Shared	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
411H	1041	IA32_MC4_STATUS	Shared	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
412H	1042	IA32_MC4_ADDR	Shared	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
				The MSR_MC4_ADDR register is either not implemented or contains no address if the ADDRV flag in the MSR_MC4_STATUS register is clear.
				When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
480H	1152	IA32_VMX_BASIC	Unique	Reporting Register of Basic VMX Capabilities (R/0) See Table 2-2.
				See Appendix A.1, "Basic VMX Information."
481H	1153	IA32_VMX_PINBASED_CTLS	Unique	Capability Reporting Register of Pin-Based VM-Execution Controls (R/O) See Table 2-2.
				See Appendix A.3, "VM-Execution Controls."
482H	1154	IA32_VMX_PROCBASED_CTLS	Unique	Capability Reporting Register of Primary Processor-Based VM-Execution Controls (R/O)
				See Appendix A.3, "VM-Execution Controls."
483H	1155	IA32_VMX_EXIT_CTLS	Unique	Capability Reporting Register of VM-Exit Controls (R/O)
				See Table 2-2.
				See Appendix A.4, "VM-Exit Controls."
484H	1156	IA32_VMX_ENTRY_CTLS	Unique	Capability Reporting Register of VM-Entry Controls (R/O) See Table 2-2.
				See Appendix A.5, "VM-Entry Controls."
485H	1157	IA32_VMX_MISC	Unique	Reporting Register of Miscellaneous VMX Capabilities (R/O)
				See Table 2-2.
				See Appendix A.6, "Miscellaneous Data."
486H	1158	IA32_VMX_CR0_FIXED0	Unique	Capability Reporting Register of CRO Bits Fixed to 0 (R/O)
				See Table 2-2.
				See Appendix A.7, "VMX-Fixed Bits in CRO."
487H	1159	IA32_VMX_CR0_FIXED1	Unique	Capability Reporting Register of CRO Bits Fixed to 1 (R/O) See Table 2-2.
				See Appendix A.7, "VMX-Fixed Bits in CR0."
488H	1160	IA32_VMX_CR4_FIXED0	Unique	Capability Reporting Register of CR4 Bits Fixed to 0 (R/0)
				See Table 2-2.
105::	4451	1000 1004 5045		See Appendix A.8, "VMX-Fixed Bits in CR4."
489H	1161	IA32_VMX_CR4_FIXED1	Unique	Capability Reporting Register of CR4 Bits Fixed to 1 (R/O) See Table 2-2.
				See Appendix A.8, "VMX-Fixed Bits in CR4."
				See Appendix A.O. VI IX I IXed DIG III CIXT.

Table 2-4. MSRs in 45 nm and 32 nm Intel Atom® Processor Family (Contd.)

Register Address		Register Name / Bit Fields	Shared/ Unique	Bit Description
Hex	Dec			
48AH	1162	IA32_VMX_VMCS_ENUM	Unique	Capability Reporting Register of VMCS Field Enumeration (R/O)
				See Table 2-2.
				See Appendix A.9, "VMCS Enumeration."
48BH	1163	IA32_VMX_PROCBASED_CTLS2	Unique	Capability Reporting Register of Secondary Processor- Based VM-Execution Controls (R/O)
				See Appendix A.3, "VM-Execution Controls."
600H	1536	IA32_DS_AREA	Unique	DS Save Area (R/W)
				See Table 2-2.
				See Section 20.6.3.4, "Debug Store (DS) Mechanism."
C000_		IA32_EFER	Unique	Extended Feature Enables
0080H				See Table 2-2.
C000_		IA32_STAR	Unique	System Call Target Address (R/W)
0081H				See Table 2-2.
C000_		IA32_LSTAR	Unique	IA-32e Mode System Call Target Address (R/W)
0082H				See Table 2-2.
C000_		IA32_FMASK	Unique	System Call Flag Mask (R/W)
0084H				See Table 2-2.
C000_		IA32_FS_BASE	Unique	Map of BASE Address of FS (R/W)
0100H				See Table 2-2.
C000_		IA32_GS_BASE	Unique	Map of BASE Address of GS (R/W)
0101H				See Table 2-2.
C000_		IA32_KERNEL_GS_BASE	Unique	Swap Target of BASE Address of GS (R/W)
0102H				See Table 2-2.

Table 2-5 lists model-specific registers (MSRs) that are specific to Intel Atom $^{(\!0\!)}$ processor with the CPUID signature with DisplayFamily_DisplayModel of 06_27H.

Table 2-5. MSRs Supported by Intel Atom® Processors with CPUID Signature 06_27H

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
3F8H	1016	MSR_PKG_C2_RESIDENCY	Package	Package C2 Residency
				Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		63:0	Package	Package C2 Residency Counter (R/O)
				Time that this package is in processor-specific C2 states since last reset. Counts at 1 Mhz frequency.

	Table 2-5. MSRS Supported by Intel Atom Processors (Conta.)with CPOID Signature 06_27H							
Register Address		Register Name / Bit Fields	Scope	Bit Description				
Hex	Dec							
3F9H	1017	MSR_PKG_C4_RESIDENCY	Package	Package C4 Residency				
				Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.				
		63:0	Package	Package C4 Residency Counter. (R/O)				
				Time that this package is in processor-specific C4 states since last reset. Counts at 1 Mhz frequency.				
3FAH	1018	MSR_PKG_C6_RESIDENCY	Package	Package C6 Residency				
				Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.				
		63:0	Package	Package C6 Residency Counter. (R/O)				

Table 2-5. MSRs Supported by Intel Atom® Processors (Contd.)with CPUID Signature 06_27H

2.4 MSRS IN INTEL PROCESSORS BASED ON SILVERMONT MICROARCHITECTURE

Table 2-6 lists model-specific registers (MSRs) common to Intel processors based on the Silvermont microarchitecture. These processors have a CPUID signature with DisplayFamily_DisplayModel of 06_37H, 06_4AH, 06_4DH, 06_5AH, and 06_5DH; see Table 2-1. The MSRs listed in Table 2-6 are also common to processors based on the Airmont microarchitecture and newer microarchitectures for next generation Intel Atom processors.

Time that this package is in processor-specific C6 states

since last reset. Counts at 1 Mhz frequency.

Table 2-7 lists MSRs common to processors based on the Silvermont and Airmont microarchitectures, but not newer microarchitectures.

Table 2-9, Table 2-9, and Table 2-10 lists MSRs that are model-specific across processors based on the Silvermont microarchitecture.

In the Silvermont microarchitecture, the scope column indicates the following: "Core" means each processor core has a separate MSR, or a bit field not shared with another processor core. "Module" means the MSR or the bit field is shared by a subset of the processor cores in the physical package. The number of processor cores in this subset is model specific and may differ between different processors. For all processors based on Silvermont microarchitecture, the L2 cache is also shared between cores in a module and thus CPUID leaf 04H enumeration can be used to figure out which processors are in the same module. "Package" means all processor cores in the physical package share the same MSR or bit interface.

Table 2-6.	MSRs Common to the Silvermont Microarchitecture and		
Newer Microarchitectures for Intel Atom® Processors			

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
ОН	0	IA32_P5_MC_ADDR	Module	See Section 2.23, "MSRs in Pentium Processors."
1H	1	IA32_P5_MC_TYPE	Module	See Section 2.23, "MSRs in Pentium Processors."
6H	6	IA32_MONITOR_FILTER_SIZE	Соге	See Section 9.10.5, "Monitor/Mwait Address Range Determination." and Table 2-2.

Table 2-6. MSRs Common to the Silvermont Microarchitecture and Newer Microarchitectures for Intel Atom® Processors

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec]		
10H	16	IA32_TIME_STAMP_COUNTER	Соге	See Section 18.17, "Time-Stamp Counter," and Table 2-2.
1BH	27	IA32_APIC_BASE	Core	See Section 11.4.4, "Local APIC Status and Location," and Table 2-2.
2AH	42	MSR_EBL_CR_POWERON	Module	Processor Hard Power-On Configuration (R/W) Writes ignored.
		63:0		Reserved
34H	52	MSR_SMI_COUNT	Core	SMI Counter (R/O)
		31:0		SMI Count (R/O) Running count of SMI events since last RESET.
		63:32		Reserved
79H	121	IA32_BIOS_UPDT_TRIG	Core	BIOS Update Trigger Register (W) See Table 2-2.
8BH	139	IA32_BIOS_SIGN_ID	Соге	BIOS Update Signature ID (R/W)
				See Table 2-2.
C1H	193	IA32_PMC0	Соге	Performance counter register
				See Table 2-2.
C2H	194	IA32_PMC1	Соге	Performance Counter Register
				See Table 2-2.
E4H	228	MSR_PMG_IO_CAPTURE_BASE	Module	Power Management IO Redirection in C-state (R/W) See http://biosbits.org.
		15:0		LVL_2 Base Address (R/W)
				Specifies the base address visible to software for IO redirection. If IO MWAIT Redirection is enabled, reads to this address will be consumed by the power management logic and decoded to MWAIT instructions. When IO port address redirection is enabled, this is the IO port address reported to the OS/software.
		18:16		C-state Range (R/W)
				Specifies the encoding value of the maximum C-State code name to be included when IO read to MWAIT redirection is enabled by MSR_PKG_CST_CONFIG_CONTROL[bit10]:
				100b - C4 is the max C-State to include
				110b - C6 is the max C-State to include 111b - C7 is the max C-State to include
		C2:10		
	22.	63:19		Reserved
E7H	231	IA32_MPERF	Core	Maximum Performance Frequency Clock Count (R/W) See Table 2-2.

Table 2-6. MSRs Common to the Silvermont Microarchitecture and Newer Microarchitectures for Intel Atom® Processors

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
E8H	232	IA32_APERF	Core	Actual Performance Frequency Clock Count (R/W)
				See Table 2-2.
FEH	254	IA32_MTRRCAP	Core	Memory Type Range Register (R)
				See Table 2-2.
13CH	316	MSR_FEATURE_CONFIG	Core	AES Configuration (RW-L)
				Privileged post-BIOS agent must provide a #GP handler to handle unsuccessful read of this MSR.
		1:0		AES Configuration (RW-L)
				Upon a successful read of this MSR, the configuration of AES instruction sets availability is as follows:
				11b: AES instructions are not available until next RESET.
				Otherwise, AES instructions are available.
				Note: AES instruction set is not available if read is unsuccessful. If the configuration is not 01b, AES instructions can be mis-configured if a privileged agent unintentionally writes 11b.
		63:2		Reserved
174H	372	IA32_SYSENTER_CS	Соге	See Table 2-2.
175H	373	IA32_SYSENTER_ESP	Соге	See Table 2-2.
176H	374	IA32_SYSENTER_EIP	Core	See Table 2-2.
179H	377	IA32_MCG_CAP	Core	See Table 2-2.
17AH	378	IA32_MCG_STATUS	Core	Global Machine Check Status
		0		RIPV When set, bit indicates that the instruction addressed by the instruction pointer pushed on the stack (when the machine check was generated) can be used to restart the program. If cleared, the program cannot be reliably restarted.
		1		EIPV
				When set, bit indicates that the instruction addressed by the instruction pointer pushed on the stack (when the machine check was generated) is directly associated with the error.
		2		MCIP
				When set, bit indicates that a machine check has been generated. If a second machine check is detected while this bit is still set, the processor enters a shutdown state. Software should write this bit to 0 after processing a machine check exception.
		63:3		Reserved
186H	390	IA32_PERFEVTSEL0	Core	See Table 2-2.
		7:0		Event Select

Table 2-6. MSRs Common to the Silvermont Microarchitecture and Newer Microarchitectures for Intel Atom® Processors

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		15:8		UMask
		16		USR
		17		OS
		18		Edge
		19		PC
		20		INT
		21		Reserved
		22		EN
		23		INV
		31:24		CMASK
		63:32		Reserved
187H	391	IA32_PERFEVTSEL1	Соге	See Table 2-2.
198H	408	IA32_PERF_STATUS	Module	See Table 2-2.
199H	409	IA32_PERF_CTL	Соге	See Table 2-2.
19AH	410	IA32_CLOCK_MODULATION	Соге	Clock Modulation (R/W)
				See Table 2-2.
				IA32_CLOCK_MODULATION MSR was originally named IA32_THERM_CONTROL MSR.
19BH	411	IA32_THERM_INTERRUPT	Core	Thermal Interrupt Control (R/W)
				See Table 2-2.
19CH	412	IA32_THERM_STATUS	Соге	Thermal Monitor Status (R/W)
				See Table 2-2.
1A2H	418	MSR_TEMPERATURE_TARGET	Package	Temperature Target
		15:0		Reserved
		23:16		Temperature Target (R)
				The default thermal throttling or PROCHOT# activation temperature in degrees C. The effective temperature for thermal throttling or PROCHOT# activation is "Temperature Target" + "Target Offset".
		29:24		Target Offset (R/W)
				Specifies an offset in degrees C to adjust the throttling and PROCHOT# activation temperature from the default target specified in TEMPERATURE_TARGET (bits 23:16).
		63:30		Reserved
1A6H	422	MSR_OFFCORE_RSP_0	Module	Offcore Response Event Select Register (R/W)
1A7H	423	MSR_OFFCORE_RSP_1	Module	Offcore Response Event Select Register (R/W)
1B0H	432	IA32_ENERGY_PERF_BIAS	Соге	See Table 2-2.

Table 2-6. MSRs Common to the Silvermont Microarchitecture and Newer Microarchitectures for Intel Atom® Processors

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
1D9H	473	IA32_DEBUGCTL	Core	Debug Control (R/W) See Table 2-2.
1DDH	477	MSR_LER_FROM_LIP	Core	Last Exception Record From Linear IP (R/W) Contains a pointer to the last branch instruction that the processor executed prior to the last exception that was generated or the last interrupt that was handled.
1DEH	478	MSR_LER_TO_LIP	Core	Last Exception Record To Linear IP (R/W) This area contains a pointer to the target of the last branch instruction that the processor executed prior to the last exception that was generated or the last interrupt that was handled.
1F2H	498	IA32_SMRR_PHYSBASE	Соге	See Table 2-2.
1F3H	499	IA32_SMRR_PHYSMASK	Соге	See Table 2-2.
200H	512	IA32_MTRR_PHYSBASE0	Соге	See Table 2-2.
201H	513	IA32_MTRR_PHYSMASK0	Соге	See Table 2-2.
202H	514	IA32_MTRR_PHYSBASE1	Core	See Table 2-2.
203H	515	IA32_MTRR_PHYSMASK1	Соге	See Table 2-2.
204H	516	IA32_MTRR_PHYSBASE2	Соге	See Table 2-2.
205H	517	IA32_MTRR_PHYSMASK2	Соге	See Table 2-2.
206H	518	IA32_MTRR_PHYSBASE3	Core	See Table 2-2.
207H	519	IA32_MTRR_PHYSMASK3	Соге	See Table 2-2.
208H	520	IA32_MTRR_PHYSBASE4	Соге	See Table 2-2.
209H	521	IA32_MTRR_PHYSMASK4	Core	See Table 2-2.
20AH	522	IA32_MTRR_PHYSBASE5	Core	See Table 2-2.
20BH	523	IA32_MTRR_PHYSMASK5	Соге	See Table 2-2.
20CH	524	IA32_MTRR_PHYSBASE6	Соге	See Table 2-2.
20DH	525	IA32_MTRR_PHYSMASK6	Core	See Table 2-2.
20EH	526	IA32_MTRR_PHYSBASE7	Core	See Table 2-2.
20FH	527	IA32_MTRR_PHYSMASK7	Соге	See Table 2-2.
250H	592	IA32_MTRR_FIX64K_00000	Соге	See Table 2-2.
258H	600	IA32_MTRR_FIX16K_80000	Соге	See Table 2-2.
259H	601	IA32_MTRR_FIX16K_A0000	Core	See Table 2-2.
268H	616	IA32_MTRR_FIX4K_C0000	Соге	See Table 2-2.
269H	617	IA32_MTRR_FIX4K_C8000	Core	See Table 2-2.
26AH	618	IA32_MTRR_FIX4K_D0000	Core	See Table 2-2.
26BH	619	IA32_MTRR_FIX4K_D8000	Core	See Table 2-2.
26CH	620	IA32_MTRR_FIX4K_E0000	Core	See Table 2-2.
26DH	621	IA32_MTRR_FIX4K_E8000	Core	See Table 2-2.

Table 2-6. MSRs Common to the Silvermont Microarchitecture and Newer Microarchitectures for Intel Atom® Processors

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
26EH	622	IA32_MTRR_FIX4K_F0000	Соге	See Table 2-2.
26FH	623	IA32_MTRR_FIX4K_F8000	Соге	See Table 2-2.
277H	631	IA32_PAT	Соге	See Table 2-2.
2FFH	767	IA32_MTRR_DEF_TYPE	Core	Default Memory Types (R/W) See Table 2-2.
309H	777	IA32_FIXED_CTR0	Core	Fixed-Function Performance Counter Register 0 (R/W) See Table 2-2.
30AH	778	IA32_FIXED_CTR1	Core	Fixed-Function Performance Counter Register 1 (R/W) See Table 2-2.
30BH	779	IA32_FIXED_CTR2	Core	Fixed-Function Performance Counter Register 2 (R/W) See Table 2-2.
345H	837	IA32_PERF_CAPABILITIES	Core	See Table 2-2. See Section 18.4.1, "IA32_DEBUGCTL MSR."
38DH	909	IA32_FIXED_CTR_CTRL	Core	Fixed-Function-Counter Control Register (R/W) See Table 2-2.
38FH	911	IA32_PERF_GLOBAL_CTRL	Core	See Table 2-2. See Section 20.6.2.2, "Global Counter Control Facilities."
3FDH	1021	MSR_CORE_C6_RESIDENCY	Core	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		63:0		CORE C6 Residency Counter (R/O)
				Value since last reset that this core is in processor- specific C6 states. Counts at the TSC Frequency.
400H	1024	IA32_MC0_CTL	Module	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
401H	1025	IA32_MC0_STATUS	Module	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
402H	1026	IA32_MCO_ADDR	Module	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs." The IA32_MCO_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MCO_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
404H	1028	IA32_MC1_CTL	Module	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
405H	1029	IA32_MC1_STATUS	Module	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
408H	1032	IA32_MC2_CTL	Module	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
409H	1033	IA32_MC2_STATUS	Module	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."

Table 2-6. MSRs Common to the Silvermont Microarchitecture and Newer Microarchitectures for Intel Atom® Processors

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
40AH	1034	IA32_MC2_ADDR	Module	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs." The IA32_MC2_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MC2_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection
4001	1026	IADD MCD CTI	Coro	exception.
40CH	1036	IA32_MC3_CTL	Core	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
40DH	1037	IA32_MC3_STATUS	Core	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
40EH	1038	IA32_MC3_ADDR	Core	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs." The MSR_MC3_ADDR register is either not implemented or contains no address if the ADDRV flag in the MSR_MC3_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
410H	1040	IA32_MC4_CTL	Core	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
411H	1041	IA32_MC4_STATUS	Core	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
412H	1042	IA32_MC4_ADDR	Core	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs." The MSR_MC4_ADDR register is either not implemented or contains no address if the ADDRV flag in the MSR_MC4_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
414H	1044	IA32_MC5_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
415H	1045	IA32_MC5_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
416H	1046	IA32_MC5_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs." The MSR_MC4_ADDR register is either not implemented or contains no address if the ADDRV flag in the MSR_MC4_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
480H	1152	IA32_VMX_BASIC	Core	Reporting Register of Basic VMX Capabilities (R/O) See Table 2-2. See Appendix A.1, "Basic VMX Information."
481H	1153	IA32_VMX_PINBASED_CTLS	Core	Capability Reporting Register of Pin-Based VM-Execution Controls (R/O) See Table 2-2. See Appendix A.3, "VM-Execution Controls."

Table 2-6. MSRs Common to the Silvermont Microarchitecture and Newer Microarchitectures for Intel Atom® Processors

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
482H	1154	IA32_VMX_PROCBASED_CTLS	Соге	Capability Reporting Register of Primary Processor- Based VM-Execution Controls (R/O)
				See Appendix A.3, "VM-Execution Controls."
483H	1155	IA32_VMX_EXIT_CTLS	Core	Capability Reporting Register of VM-Exit Controls (R/O)
				See Table 2-2.
				See Appendix A.4, "VM-Exit Controls."
484H	1156	IA32_VMX_ENTRY_CTLS	Соге	Capability Reporting Register of VM-Entry Controls (R/O)
				See Table 2-2.
				See Appendix A.5, "VM-Entry Controls."
485H	1157	IA32_VMX_MISC	Core	Reporting Register of Miscellaneous VMX Capabilities (R/O)
				See Table 2-2.
				See Appendix A.6, "Miscellaneous Data."
486H	1158	IA32_VMX_CRO_FIXEDO	Core	Capability Reporting Register of CRO Bits Fixed to 0 (R/O)
				See Table 2-2.
				See Appendix A.7, "VMX-Fixed Bits in CR0."
487H	1159	IA32_VMX_CRO_FIXED1	Core	Capability Reporting Register of CRO Bits Fixed to 1 (R/O)
				See Table 2-2.
				See Appendix A.7, "VMX-Fixed Bits in CR0."
488H	1160	IA32_VMX_CR4_FIXED0	Core	Capability Reporting Register of CR4 Bits Fixed to 0 (R/0)
				See Table 2-2.
				See Appendix A.8, "VMX-Fixed Bits in CR4."
489H	1161	IA32_VMX_CR4_FIXED1	Core	Capability Reporting Register of CR4 Bits Fixed to 1 (R/0)
				See Table 2-2.
				See Appendix A.8, "VMX-Fixed Bits in CR4."
48AH	1162	IA32_VMX_VMCS_ENUM	Соге	Capability Reporting Register of VMCS Field Enumeration (R/O)
				See Table 2-2.
				See Appendix A.9, "VMCS Enumeration."
48BH	1163	IA32_VMX_PROCBASED_CTLS2	Соге	Capability Reporting Register of Secondary Processor- Based VM-Execution Controls (R/O)
				See Appendix A.3, "VM-Execution Controls."
48CH	1164	IA32_VMX_EPT_VPID_ENUM	Core	Capability Reporting Register of EPT and VPID (R/O) See Table 2-2
48DH	1165	IA32_VMX_TRUE_PINBASED_CTLS	Core	Capability Reporting Register of Pin-Based VM-Execution Flex Controls (R/O)
				See Table 2-2

Table 2-6. MSRs Common to the Silvermont Microarchitecture and Newer Microarchitectures for Intel Atom® Processors

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
48EH	1166	IA32_VMX_TRUE_PROCBASED_CTLS	Core	Capability Reporting Register of Primary Processor- based VM-Execution Flex Controls (R/O)
				See Table 2-2
48FH	1167	IA32_VMX_TRUE_EXIT_CTLS	Core	Capability Reporting Register of VM-Exit Flex Controls (R/O)
				See Table 2-2
490H	1168	IA32_VMX_TRUE_ENTRY_CTLS	Core	Capability Reporting Register of VM-Entry Flex Controls (R/O)
				See Table 2-2
491H	1169	IA32_VMX_FMFUNC	Соге	Capability Reporting Register of VM-Function Controls (R/O)
				See Table 2-2
4C1H	1217	IA32_A_PMC0	Core	See Table 2-2.
4C2H	1218	IA32_A_PMC1	Core	See Table 2-2.
600H	1536	IA32_DS_AREA	Core	DS Save Area (R/W)
				See Table 2-2.
				See Section 20.6.3.4, "Debug Store (DS) Mechanism."
660H	1632	MSR_CORE_C1_RESIDENCY	Core	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		63:0		CORE C1 Residency Counter. (R/O)
				Value since last reset that this core is in processor- specific C1 states. Counts at the TSC frequency.
6E0H	1760	IA32_TSC_DEADLINE	Core	TSC Target of Local APIC's TSC Deadline Mode (R/W) See Table 2-2.
C000_		IA32_EFER	Соге	Extended Feature Enables
0080H		_		See Table 2-2.
C000_ 0081H		IA32_STAR	Core	System Call Target Address (R/W) See Table 2-2.
C000_		IA32_LSTAR	Core	IA-32e Mode System Call Target Address (R/W)
0082H		_		See Table 2-2.
C000_		IA32_FMASK	Соге	System Call Flag Mask (R/W)
0084H				See Table 2-2.
C000_		IA32_FS_BASE	Core	Map of BASE Address of FS (R/W)
0100H				See Table 2-2.
C000_		IA32_GS_BASE	Core	Map of BASE Address of GS (R/W)
0101H				See Table 2-2.
C000_ 0102H		IA32_KERNEL_GS_BASE	Core	Swap Target of BASE Address of GS (R/W) See Table 2-2.

Table 2-6. MSRs Common to the Silvermont Microarchitecture and Newer Microarchitectures for Intel Atom® Processors

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
C000_		IA32_TSC_AUX	Core	AUXILIARY TSC Signature (R/W)
0103H				See Table 2-2

Table 2-7 lists model-specific registers (MSRs) that are common to Intel Atom $^{(\!R\!)}$ processors based on the Silvermont and Airmont microarchitectures but not newer microarchitectures.

Table 2-7. MSRs Common to the Silvermont and Airmont Microarchitectures

Regi Add	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
17H	23	MSR_PLATFORM_ID	Module	Model Specific Platform ID (R)
		7:0		Reserved
		13:8		Maximum Qualified Ratio (R)
				The maximum allowed bus ratio.
		49:13		Reserved
		52:50		See Table 2-2.
		63:33		Reserved
ЗАН	58	IA32_FEATURE_CONTROL	Соге	Control Features in Intel 64Processor (R/W)
				See Table 2-2.
		0		Lock (R/WL)
		1		Reserved
		2		Enable VMX outside SMX operation (R/WL)
40H	64	MSR_LASTBRANCH_0_FROM_IP	Соге	Last Branch Record O From IP (R/W)
				One of eight pairs of last branch record registers on the last branch record stack. The From_IP part of the stack contains pointers to the source instruction. See also:
				Last Branch Record Stack TOS at 1C9H.Section 18.5 and record format in Section 18.4.8.1.
41H	65	MSR_LASTBRANCH_1_FROM_IP	Соге	Last Branch Record 1 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
42H	66	MSR_LASTBRANCH_2_FROM_IP	Соге	Last Branch Record 2 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
43H	67	MSR_LASTBRANCH_3_FROM_IP	Соге	Last Branch Record 3 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
44H	68	MSR_LASTBRANCH_4_FROM_IP	Core	Last Branch Record 4 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
45H	69	MSR_LASTBRANCH_5_FROM_IP	Core	Last Branch Record 5 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
46H	70	MSR_LASTBRANCH_6_FROM_IP	Соге	Last Branch Record 6 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.

Table 2-7. MSRs Common to the Silvermont and Airmont Microarchitectures

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
47H	71	MSR_LASTBRANCH_7_FROM_IP	Core	Last Branch Record 7 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
60H	96	MSR_LASTBRANCH_O_TO_IP	Core	Last Branch Record 0 To IP (R/W) One of eight pairs of last branch record registers on the last branch record stack. The To_IP part of the stack contains pointers to the destination instruction.
61H	97	MSR_LASTBRANCH_1_TO_IP	Core	Last Branch Record 1 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
62H	98	MSR_LASTBRANCH_2_TO_IP	Core	Last Branch Record 2 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
63H	99	MSR_LASTBRANCH_3_TO_IP	Core	Last Branch Record 3 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
64H	100	MSR_LASTBRANCH_4_TO_IP	Core	Last Branch Record 4 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
65H	101	MSR_LASTBRANCH_5_TO_IP	Core	Last Branch Record 5 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
66H	102	MSR_LASTBRANCH_6_TO_IP	Core	Last Branch Record 6 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
67H	103	MSR_LASTBRANCH_7_TO_IP	Core	Last Branch Record 7 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
CEH	206	MSR_PLATFORM_INFO	Package	Platform Information: Contains power management and other model specific features enumeration. See http://biosbits.org.
		7:0		Reserved
		15:8	Package	Maximum Non-Turbo Ratio (R/O) This is the ratio of the maximum frequency that does not require turbo. Frequency = ratio * Scalable Bus Frequency.
		63:16		Reserved
E2H	226	MSR_PKG_CST_CONFIG_CONTROL	Module	C-State Configuration Control (R/W) Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States. See http://biosbits.org.

Table 2-7. MSRs Common to the Silvermont and Airmont Microarchitectures

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		2:0		Package C-State Limit (R/W) Specifies the lowest processor-specific C-state code name (consuming the least power) for the package. The default is set as factory-configured package C-state limit. The following C-state code name encodings are supported: 000b: C0 (no package C-sate support) 001b: C1 (Behavior is the same as 000b) 100b: C4 110b: C6 111b: C7 (Silvermont only).
		9:3		Reserved
		10		I/O MWAIT Redirection Enable (R/W) When set, will map IO_read instructions sent to IO register specified by MSR_PMG_IO_CAPTURE_BASE to MWAIT instructions.
		14:11		Reserved
		15		CFG Lock (R/WO) When set, locks bits 15:0 of this register until next reset.
		63:16		Reserved
11EH	281	MSR_BBL_CR_CTL3	Module	Control Register 3 Used to configure the L2 Cache.
		0		L2 Hardware Enabled (R/O) 1 = If the L2 is hardware-enabled. 0 = Indicates if the L2 is hardware-disabled.
		7:1		Reserved
		8		L2 Enabled (R/W) 1 = L2 cache has been initialized. 0 = Disabled (default). Until this bit is set the processor will not respond to the WBINVD instruction or the assertion of the FLUSH# input.
		22:9		Reserved
		23		L2 Not Present (R/0) 0 = L2 Present. 1 = L2 Not Present.
		63:24		Reserved
1A0H	416	IA32_MISC_ENABLE		Enable Misc. Processor Features (R/W) Allows a variety of processor functions to be enabled and disabled.
		0	Core	Fast-Strings Enable See Table 2-2.

Table 2-7. MSRs Common to the Silvermont and Airmont Microarchitectures

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		2:1		Reserved
		3	Module	Automatic Thermal Control Circuit Enable (R/W) See Table 2-2. Default value is 0.
		6:4		Reserved
		7	Core	Performance Monitoring Available (R) See Table 2-2.
		10:8		Reserved
		11	Core	Branch Trace Storage Unavailable (R/O) See Table 2-2.
		12	Core	Processor Event Based Sampling Unavailable (R/O) See Table 2-2.
		15:13		Reserved
		16	Module	Enhanced Intel SpeedStep Technology Enable (R/W) See Table 2-2.
		18	Core	ENABLE MONITOR FSM (R/W) See Table 2-2.
		21:19		Reserved
		22	Core	Limit CPUID Maxval (R/W) See Table 2-2.
		23	Module	xTPR Message Disable (R/W) See Table 2-2.
		33:24		Reserved
		34	Core	XD Bit Disable (R/W) See Table 2-2.
		37:35		Reserved
		38	Module	Turbo Mode Disable (R/W)
				When set to 1 on processors that support Intel Turbo Boost Technology, the turbo mode feature is disabled and the IDA_Enable feature flag will be cleared (CPUID.06H: EAX[1]=0).
				When set to a 0 on processors that support IDA, CPUID.06H: EAX[1] reports the processor's support of turbo mode is enabled.
				Note: The power-on default value is used by BIOS to detect hardware support of turbo mode. If the power-on default value is 1, turbo mode is available in the processor. If the power-on default value is 0, turbo mode is not available.
		63:39		Reserved
1C8H	456	MSR_LBR_SELECT	Core	Last Branch Record Filtering Select Register (R/W) See Section 18.9.2, "Filtering of Last Branch Records."

Table 2-7. MSRs Common to the Silvermont and Airmont Microarchitectures

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		0		CPL_EQ_0
		1		CPL_NEQ_0
		2		JCC
		3		NEAR_REL_CALL
		4		NEAR_IND_CALL
		5		NEAR_RET
		6		NEAR_IND_JMP
		7		NEAR_REL_JMP
		8		FAR_BRANCH
		63:9		Reserved
1C9H	457	MSR_LASTBRANCH_TOS	Core	Last Branch Record Stack TOS (R/W)
				Contains an index (bits 0-2) that points to the MSR containing the most recent branch record.
				See MSR_LASTBRANCH_O_FROM_IP.
38EH	910	IA32_PERF_GLOBAL_STATUS	Core	See Table 2-2. See Section 20.6.2.2, "Global Counter Control Facilities."
390H	912	IA32_PERF_GLOBAL_OVF_CTRL	Core	See Table 2-2. See Section 20.6.2.2, "Global Counter Control Facilities."
3F1H	1009	IA32_PEBS_ENABLE	Core	See Table 2-2. See Section 20.6.2.4, "Processor Event
		(MSR_PEBS_ENABLE)		Based Sampling (PEBS)."
		0		Enable PEBS for precise event on IA32_PMC0 (R/W)
3FAH	1018	MSR_PKG_C6_RESIDENCY	Package	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		63:0		Package C6 Residency Counter (R/O)
				Value since last reset that this package is in processor- specific C6 states. Counts at the TSC Frequency.
664H	1636	MSR_MC6_RESIDENCY_COUNTER	Module	Module C6 Residency Counter (R/0)
				Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		63:0		Time that this module is in module-specific C6 states since last reset. Counts at 1 Mhz frequency.

2.4.1 MSRs with Model-Specific Behavior in the Silvermont Microarchitecture

Table 2-8 lists model-specific registers (MSRs) that are specific to Intel Atom $^{\circledR}$ processor E3000 Series (CPUID signature with DisplayFamily_DisplayModel of 06_37H) and Intel Atom processors (CPUID signatures with DisplayFamily_DisplayModel of 06_4AH, 06_5AH, 06_5DH).

Table 2-8. Specific MSRs Supported by Intel Atom $^\circ$ Processors with CPUID Signatures 06_37H, 06_4AH, 06_5AH, 06_5DH

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
CDH	205	MSR_FSB_FREQ	Module	Scaleable Bus Speed (R/O) This field indicates the intended scaleable bus clock speed for processors based on Silvermont microarchitecture.
		2:0		 100B: 080.0 MHz 000B: 083.3 MHz 001B: 100.0 MHz 010B: 133.3 MHz 011B: 116.7 MHz
		63:3		Reserved
606H	1542	MSR_RAPL_POWER_UNIT	Package	Unit Multipliers used in RAPL Interfaces (R/O)
				See Section 15.10.1, "RAPL Interfaces."
		3:0		Power Units
				Power related information (in milliWatts) is based on the multiplier, 2^PU; where PU is an unsigned integer represented by bits 3:0. Default value is 0101b, indicating power unit is in 32 milliWatts increment.
		7:4		Reserved
		12:8		Energy Status Units
				Energy related information (in microJoules) is based on the multiplier, 2^ESU; where ESU is an unsigned integer represented by bits 12:8. Default value is 00101b, indicating energy unit is in 32 microJoules increment.
		15:13		Reserved
		19:16		Time Unit
				The value is 0000b, indicating time unit is in one second.
		63:20		Reserved
610H	1552	MSR_PKG_POWER_LIMIT	Package	PKG RAPL Power Limit Control (R/W)
		14:0		Package Power Limit #1 (R/W)
				See Section 15.10.3, "Package RAPL Domain." and MSR_RAPL_POWER_UNIT in Table 2-8.
		15		Enable Power Limit #1 (R/W)
				See Section 15.10.3, "Package RAPL Domain."
		16		Package Clamping Limitation #1 (R/W)
				See Section 15.10.3, "Package RAPL Domain."
		23:17		Time Window for Power Limit #1 (R/W)
				In unit of second. If 0 is specified in bits [23:17], defaults to 1 second window.
		63:24		Reserved
611H	1553	MSR_PKG_ENERGY_STATUS	Package	PKG Energy Status (R/0)
				See Section 15.10.3, "Package RAPL Domain." and MSR_RAPL_POWER_UNIT in Table 2-8.

Table 2-8. Specific MSRs Supported by Intel Atom® Processors with CPUID Signatures 06_37H, 06_4AH, 06_5AH, 06_5DH

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
639H	1593	MSR_PP0_ENERGY_STATUS	Package	PPO Energy Status (R/O) See Section 15.10.4, "PPO/PP1 RAPL Domains." and MSR_RAPL_POWER_UNIT in Table 2-8.

Table 2-9 lists model-specific registers (MSRs) that are specific to Intel Atom $^{(\!0)}$ processor E3000 Series (CPUID signature with DisplayFamily_DisplayModel of 06_37H).

Table 2-9. Specific MSRs Supported by Intel Atom® Processor E3000 Series with CPUID Signature 06_37H

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
668H	1640	MSR_CC6_DEMOTION_POLICY_CONFIG	Package	Core C6 Demotion Policy Config MSR
		63:0		Controls per-core C6 demotion policy. Writing a value of 0 disables core level HW demotion policy.
669H	1641	MSR_MC6_DEMOTION_POLICY_CONFIG	Package	Module C6 Demotion Policy Config MSR
		63:0		Controls module (i.e., two cores sharing the second-level cache) C6 demotion policy. Writing a value of 0 disables module level HW demotion policy.
664H	1636	MSR_MC6_RESIDENCY_COUNTER	Module	Module C6 Residency Counter (R/0)
				Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		63:0		Time that this module is in module-specific C6 states since last reset. Counts at 1 Mhz frequency.

Table 2-10 lists model-specific registers (MSRs) that are specific to Intel Atom® processor C2000 Series (CPUID signature with DisplayFamily_DisplayModel of 06_4DH).

Table 2-10. Specific MSRs Supported by Intel Atom® Processor C2000 Series with CPUID Signature 06_4DH

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
1A4H	420	MSR_MISC_FEATURE_CONTROL		Miscellaneous Feature Control (R/W)
		0	Core	L2 Hardware Prefetcher Disable (R/W) If 1, disables the L2 hardware prefetcher, which fetches additional lines of code or data into the L2 cache.
		1		Reserved
		2	Core	DCU Hardware Prefetcher Disable (R/W) If 1, disables the L1 data cache prefetcher, which fetches the next cache line into L1 data cache.
		63:3		Reserved

Table 2-10. Specific MSRs Supported by Intel Atom® Processor C2000 Series (Contd.)with CPUID Signature

_	ster ress	Register Name / Bit Fields Sc		Bit Description
Hex	Dec			
1ADH	429	MSR_TURBO_RATIO_LIMIT	Package	Maximum Ratio Limit of Turbo Mode (R/W)
		7:0	Package	Maximum Ratio Limit for 1C
				Maximum turbo ratio limit of 1 core active.
		15:8	Package	Maximum Ratio Limit for 2C
				Maximum turbo ratio limit of 2 core active.
		23:16	Package	Maximum Ratio Limit for 3C
				Maximum turbo ratio limit of 3 core active.
		31:24	Package	Maximum Ratio Limit for 4C
				Maximum turbo ratio limit of 4 core active.
		39:32	Package	Maximum Ratio Limit for 5C
				Maximum turbo ratio limit of 5 core active.
		47:40	Package	Maximum Ratio Limit for 6C
				Maximum turbo ratio limit of 6 core active.
		55:48	Package	Maximum Ratio Limit for 7C
				Maximum turbo ratio limit of 7 core active.
		63:56	Package	Maximum Ratio Limit for 8C
				Maximum turbo ratio limit of 8 core active.
606H	1542	MSR_RAPL_POWER_UNIT	Package	Unit Multipliers used in RAPL Interfaces (R/O)
				See Section 15.10.1, "RAPL Interfaces."
		3:0		Power Units
				Power related information (in milliWatts) is based on the multiplier, 2^PU; where PU is an unsigned integer
				represented by bits 3:0. Default value is 0101b, indicating power unit is in 32 milliWatts increment.
		7:4		Reserved
		12:8		Energy Status Units.
				Energy related information (in microJoules) is based on the multiplier, 2^ESU; where ESU is an unsigned integer represented by bits 12:8. Default value is 00101b, indicating energy unit is in 32 microJoules increment.
		15:13		Reserved
		19:16		Time Unit
				The value is 0000b, indicating time unit is in one second.
		63:20		Reserved
610H	1552	MSR_PKG_POWER_LIMIT	Package	PKG RAPL Power Limit Control (R/W)
				See Section 15.10.3, "Package RAPL Domain."
66EH	1646	MSR_PKG_POWER_INFO	Package	PKG RAPL Parameter (R/0)

Table 2-10. Specific MSRs Supported by Intel Atom® Processor C2000 Series (Contd.)with CPUID Signature

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		14:0		Thermal Spec Power (R/0)
				The unsigned integer value is the equivalent of the thermal specification power of the package domain. The unit of this field is specified by the "Power Units" field of MSR_RAPL_POWER_UNIT.
		63:15		Reserved

2.4.2 MSRs In Intel Atom® Processors Based on Airmont Microarchitecture

Intel Atom processor X7-Z8000 and X5-Z8000 series are based on the Airmont microarchitecture. These processors support MSRs listed in Table 2-6, Table 2-7, Table 2-8, and Table 2-11. These processors have a CPUID signature with DisplayFamily_DisplayModel including 06_4CH; see Table 2-1.

Table 2-11. MSRs in Intel Atom® Processors Based on the Airmont Microarchitecture

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
CDH	205	MSR_FSB_FREQ	Module	Scaleable Bus Speed (R/O) This field indicates the intended scaleable bus clock speed for processors based on Airmont microarchitecture.
		3:0		 0000B: 083.3 MHz 0001B: 100.0 MHz 0010B: 133.3 MHz 0011B: 116.7 MHz 0100B: 080.0 MHz 0101B: 093.3 MHz 0110B: 090.0 MHz 0111B: 088.9 MHz 1000B: 087.5 MHz
		63:5		Reserved
E2H	226	MSR_PKG_CST_CONFIG_CONTROL	Module	C-State Configuration Control (R/W) Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States. See http://biosbits.org.

Table 2-11. MSRs in Intel Atom® Processors Based on the Airmont Microarchitecture (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		2:0		Package C-State Limit (R/W) Specifies the lowest processor-specific C-state code name (consuming the least power) for the package.
				The default is set as factory-configured package C-state limit. The following C-state code name encodings are
				supported: 000b: No limit
				000b. No lifflit
				010b: C2
				110b: C6
				111b: C7
		9:3		Reserved
		10		I/O MWAIT Redirection Enable (R/W)
				When set, will map IO_read instructions sent to IO register specified by MSR_PMG_IO_CAPTURE_BASE to MWAIT instructions.
		14:11		Reserved
		15		CFG Lock (R/WO)
				When set, locks bits 15:0 of this register until next reset.
		63:16		Reserved
E4H	228	MSR_PMG_IO_CAPTURE_BASE	Module	Power Management IO Redirection in C-state (R/W) See http://biosbits.org.
		15:0		LVL_2 Base Address (R/W)
				Specifies the base address visible to software for IO redirection. If IO MWAIT Redirection is enabled, reads to this address will be consumed by the power management logic and decoded to MWAIT instructions. When IO port address redirection is enabled, this is the IO port address reported to the OS/software.
		18:16		C-state Range (R/W)
				Specifies the encoding value of the maximum C-State code name to be included when IO read to MWAIT redirection is enabled by MSR_PKG_CST_CONFIG_CONTROL[bit10]:
				000b - C3 is the max C-State to include.
				001b - Deep Power Down Technology is the max C-State.
				010b - C7 is the max C-State to include.
		63:19		Reserved
638H	1592	MSR_PPO_POWER_LIMIT	Package	PPO RAPL Power Limit Control (R/W)

Table 2-11	MSRs in Intel Atom	Processors Based	l on the Airmont	Microarchitecture (Contd)
	1,12V2 III IIIIGI VIOIII	Lincessors paser	I OII LIIE AIIIIOIIL	increase interture i	COIILU.

Regi Addı		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		14:0		PPO Power Limit #1 (R/W)
				See Section 15.10.4, "PPO/PP1 RAPL Domains" and MSR_RAPL_POWER_UNIT in Table 2-8.
		15		Enable Power Limit #1 (R/W)
				See Section 15.10.4, "PPO/PP1 RAPL Domains."
		16		Reserved
		23:17		Time Window for Power Limit #1 (R/W)
				Specifies the time duration over which the average power must remain below PPO_POWER_LIMIT #1(14:0). Supported Encodings:
				0x0: 1 second time duration.
				0x1: 5 second time duration (Default).
				0x2: 10 second time duration.
				0x3: 15 second time duration.
				0x4: 20 second time duration.
				0x5: 25 second time duration.
				0x6: 30 second time duration.
				0x7: 35 second time duration.
				0x8: 40 second time duration.
				0x9: 45 second time duration.
				0xA: 50 second time duration.
				0xB-0x7F - reserved.
		63:24		Reserved

2.5 MSRS IN INTEL ATOM® PROCESSORS BASED ON GOLDMONT MICROARCHITECTURE

Intel Atom processors based on the Goldmont microarchitecture support MSRs listed in Table 2-6 and Table 2-12. These processors have a CPUID signature with DisplayFamily_DisplayModel of 06_5CH; see Table 2-1.

In the Goldmont microarchitecture, the scope column indicates the following: "Core" means each processor core has a separate MSR, or a bit field not shared with another processor core. "Module" means the MSR or the bit field is shared by a subset of the processor cores in the physical package. The number of processor cores in this subset is model specific and may differ between different processors. For all processors based on Goldmont microarchitecture, the L2 cache is also shared between cores in a module and thus CPUID leaf 04H enumeration can be used to figure out which processors are in the same module. "Package" means all processor cores in the physical package share the same MSR or bit interface.

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
17H	23	MSR_PLATFORM_ID	Module	Model Specific Platform ID (R)

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

	ister Iress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		49:0		Reserved
		52:50		See Table 2-2.
		63:33		Reserved
ЗАН	58	IA32_FEATURE_CONTROL	Core	Control Features in Intel 64 Processor (R/W) See Table 2-2.
		0		Lock (R/WL)
		1		Enable VMX inside SMX operation (R/WL)
		2		Enable VMX outside SMX operation (R/WL)
		14:8		SENTER local functions enables (R/WL)
		15		SENTER global functions enable (R/WL)
		18		SGX global functions enable (R/WL)
		63:19		Reserved
3BH	59	IA32_TSC_ADJUST	Core	Per-Core TSC ADJUST (R/W) See Table 2-2.
СЗН	195	IA32_PMC2	Соге	Performance Counter Register See Table 2-2.
C4H	196	IA32_PMC3	Соге	Performance Counter Register See Table 2-2.
CEH	206	MSR_PLATFORM_INFO	Package	Platform Information
				Contains power management and other model specific features enumeration. See http://biosbits.org.
		7:0		Reserved
		15:8	Package	Maximum Non-Turbo Ratio (R/O)
				This is the ratio of the maximum frequency that does not require turbo. Frequency = ratio * 100 MHz.
		27:16		Reserved
		28	Package	Programmable Ratio Limit for Turbo Mode (R/O) When set to 1, indicates that Programmable Ratio Limit for Turbo mode is enabled. When set to 0, indicates Programmable Ratio Limit for Turbo mode is disabled.
		29	Package	Programmable TDP Limit for Turbo Mode (R/O)
				When set to 1, indicates that TDP Limit for Turbo mode is programmable. When set to 0, indicates TDP Limit for Turbo mode is not programmable.
		30	Package	Programmable TJ OFFSET (R/O)
				When set to 1, indicates that MSR_TEMPERATURE_TARGET.[27:24] is valid and writable to specify a temperature offset.
		39:31		Reserved

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		47:40	Package	Maximum Efficiency Ratio (R/O)
				This is the minimum ratio (maximum efficiency) that the processor can operate, in units of 100MHz.
		63:48		Reserved
E2H	226	MSR_PKG_CST_CONFIG_CONTROL	Core	C-State Configuration Control (R/W)
				Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
				See http://biosbits.org.
		3:0		Package C-State Limit (R/W)
				Specifies the lowest processor-specific C-state code name (consuming the least power) for the package. The default is set as factory-configured package C-state limit.
				The following C-state code name encodings are supported:
				0000b: No limit
				0001b: C1
				0010b: C3
				0011b: C6
				0100b: C7
				0101b: C7S
				0110b: C8
				0111b: C9
				1000b: C10
		9:3		Reserved
		10		I/O MWAIT Redirection Enable (R/W)
				When set, will map IO_read instructions sent to IO register specified by MSR_PMG_IO_CAPTURE_BASE to MWAIT instructions.
		14:11		Reserved
		15		CFG Lock (R/WO)
				When set, locks bits 15:0 of this register until next reset.
		63:16		Reserved
17DH	381	MSR_SMM_MCA_CAP	Core	Enhanced SMM Capabilities (SMM-RO)
				Reports SMM capability enhancement. Accessible only while in SMM.
		57:0		Reserved
		58		SMM_Code_Access_Chk (SMM-RO)
				If set to 1 indicates that the SMM code access restriction is supported and the MSR_SMM_FEATURE_CONTROL is supported.

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

Regi: Addr		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		59		Long_Flow_Indication (SMM-RO) If set to 1 indicates that the SMM long flow indicator is supported and the MSR_SMM_DELAYED is supported.
		63:60		Reserved
188H	392	IA32_PERFEVTSEL2	Соге	See Table 2-2.
189H	393	IA32_PERFEVTSEL3	Соге	See Table 2-2.
1A0H	416	IA32_MISC_ENABLE		Enable Misc. Processor Features (R/W)
				Allows a variety of processor functions to be enabled and disabled.
		0	Соге	Fast-Strings Enable
				See Table 2-2.
		2:1		Reserved
		3	Package	Automatic Thermal Control Circuit Enable (R/W)
				See Table 2-2. Default value is 1.
		6:4		Reserved
		7	Соге	Performance Monitoring Available (R)
				See Table 2-2.
		10:8		Reserved
		11	Соге	Branch Trace Storage Unavailable (R/O) See Table 2-2.
		12	Соге	Processor Event Based Sampling Unavailable (R/O) See Table 2-2.
		15:13		Reserved
		16	Package	Enhanced Intel SpeedStep Technology Enable (R/W) See Table 2-2.
		18	Соге	ENABLE MONITOR FSM (R/W)
				See Table 2-2.
		21:19		Reserved
		22	Соге	Limit CPUID Maxval (R/W)
				See Table 2-2.
		23	Package	xTPR Message Disable (R/W) See Table 2-2.
		33:24		Reserved
		34	Core	XD Bit Disable (R/W)
				See Table 2-2.
		37:35		Reserved

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec]		
		38	Package	Turbo Mode Disable (R/W)
				When set to 1 on processors that support Intel Turbo Boost Technology, the turbo mode feature is disabled and the IDA_Enable feature flag will be clear (CPUID.06H: EAX[1]=0).
				When set to a 0 on processors that support IDA, CPUID.06H: EAX[1] reports the processor's support of turbo mode is enabled.
				Note: The power-on default value is used by BIOS to detect hardware support of turbo mode. If the power-on default value is 1, turbo mode is available in the processor. If the power-on default value is 0, turbo mode is not available.
		63:39		Reserved
1A4H	420	MSR_MISC_FEATURE_CONTROL		Miscellaneous Feature Control (R/W)
		0	Соге	L2 Hardware Prefetcher Disable (R/W)
				If 1, disables the L2 hardware prefetcher, which fetches additional lines of code or data into the L2 cache.
		1		Reserved
		2	Core	DCU Hardware Prefetcher Disable (R/W)
				If 1, disables the L1 data cache prefetcher, which fetches the next cache line into L1 data cache.
		63:3		Reserved
1AAH	426	MSR_MISC_PWR_MGMT	Package	Miscellaneous Power Management Control
				Various model specific features enumeration. See http://biosbits.org.
		0		EIST Hardware Coordination Disable (R/W)
				When 0, enables hardware coordination of Enhanced Intel Speedstep Technology request from processor cores. When 1, disables hardware coordination of Enhanced Intel Speedstep Technology requests.
		21:1		Reserved
		22		Thermal Interrupt Coordination Enable (R/W)
				If set, then thermal interrupt on one core is routed to all cores.
		63:23		Reserved

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

Regi: Addr	ster	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
1ADH	429	MSR_TURBO_RATIO_LIMIT	Package	Maximum Ratio Limit of Turbo Mode by Core Groups (R/W)
				Specifies Maximum Ratio Limit for each Core Group. Max ratio for groups with more cores must decrease monotonically.
				For groups with less than 4 cores, the max ratio must be 32 or less. For groups with 4-5 cores, the max ratio must be 22 or less. For groups with more than 5 cores, the max ratio must be 16 or less.
		7:0	Package	Maximum Ratio Limit for Active Cores in Group 0
				Maximum turbo ratio limit when the number of active cores is less than or equal to the Group 0 threshold.
		15:8	Package	Maximum Ratio Limit for Active Cores in Group 1
				Maximum turbo ratio limit when the number of active cores is less than or equal to the Group 1 threshold, and greater than the Group 0 threshold.
		23:16	Package	Maximum Ratio Limit for Active Cores in Group 2
				Maximum turbo ratio limit when the number of active cores is less than or equal to the Group 2 threshold, and greater than the Group 1 threshold.
		31:24	Package	Maximum Ratio Limit for Active Cores in Group 3
				Maximum turbo ratio limit when the number of active cores is less than or equal to the Group 3 threshold, and greater than the Group 2 threshold.
		39:32	Package	Maximum Ratio Limit for Active Cores in Group 4
				Maximum turbo ratio limit when the number of active cores is less than or equal to the Group 4 threshold, and greater than the Group 3 threshold.
		47:40	Package	Maximum Ratio Limit for Active Cores in Group 5
				Maximum turbo ratio limit when the number of active cores is less than or equal to the Group 5 threshold, and greater than the Group 4 threshold.
		55:48	Package	Maximum Ratio Limit for Active Cores in Group 6
				Maximum turbo ratio limit when the number of active cores is less than or equal to the Group 6 threshold, and greater than the Group 5 threshold.
		63:56	Package	Maximum Ratio Limit for Active Cores in Group 7
				Maximum turbo ratio limit when the number of active cores is less than or equal to the Group 7 threshold, and greater than the Group 6 threshold.
1AEH	430	MSR_TURBO_GROUP_CORECNT	Package	Group Size of Active Cores for Turbo Mode Operation (R/W)
				Writes of 0 threshold is ignored.

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		7:0	Package	Group 0 Core Count Threshold Maximum number of active cores to operate under the Group 0 Max Turbo Ratio limit.
		15:8	Package	Group 1 Core Count Threshold Maximum number of active cores to operate under the Group 1 Max Turbo Ratio limit. Must be greater than the Group 0 Core Count.
		23:16	Package	Group 2 Core Count Threshold Maximum number of active cores to operate under the Group 2 Max Turbo Ratio limit. Must be greater than the Group 1 Core Count.
		31:24	Package	Group 3 Core Count Threshold Maximum number of active cores to operate under the Group 3 Max Turbo Ratio limit. Must be greater than the Group 2 Core Count.
		39:32	Package	Group 4 Core Count Threshold Maximum number of active cores to operate under the Group 4 Max Turbo Ratio limit. Must be greater than the Group 3 Core Count.
		47:40	Package	Group 5 Core Count Threshold Maximum number of active cores to operate under the Group 5 Max Turbo Ratio limit. Must be greater than the Group 4 Core Count.
		55:48	Package	Group 6 Core Count Threshold Maximum number of active cores to operate under the Group 6 Max Turbo Ratio limit. Must be greater than the Group 5 Core Count.
		63:56	Package	Group 7 Core Count Threshold Maximum number of active cores to operate under the Group 7 Max Turbo Ratio limit. Must be greater than the Group 6 Core Count, and not less than the total number of processor cores in the package. E.g., specify 255.
1C8H	456	MSR_LBR_SELECT	Core	Last Branch Record Filtering Select Register (R/W) See Section 18.9.2, "Filtering of Last Branch Records."
		0		CPL_EQ_0
		1		CPL_NEQ_0
		2		JCC
		3		NEAR_REL_CALL
		4		NEAR_IND_CALL
		5		NEAR_RET
		6		NEAR_IND_JMP
		7		NEAR_REL_JMP

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		8		FAR_BRANCH
		9		EN_CALL_STACK
		63:10		Reserved
1C9H	457	MSR_LASTBRANCH_TOS	Соге	Last Branch Record Stack TOS (R/W)
				Contains an index (bits 0-4) that points to the MSR containing the most recent branch record.
				See MSR_LASTBRANCH_0_FROM_IP.
1FCH	508	MSR_POWER_CTL	Соге	Power Control Register. See http://biosbits.org.
		0		Reserved
		1	Package	C1E Enable (R/W)
				When set to '1', will enable the CPU to switch to the Minimum Enhanced Intel SpeedStep Technology operating point when all execution cores enter MWAIT (C1).
		63:2		Reserved
210H	528	IA32_MTRR_PHYSBASE8	Соге	See Table 2-2.
211H	529	IA32_MTRR_PHYSMASK8	Соге	See Table 2-2.
212H	530	IA32_MTRR_PHYSBASE9	Соге	See Table 2-2.
213H	531	IA32_MTRR_PHYSMASK9	Соге	See Table 2-2.
280H	640	IA32_MC0_CTL2	Module	See Table 2-2.
281H	641	IA32_MC1_CTL2	Module	See Table 2-2.
282H	642	IA32_MC2_CTL2	Соге	See Table 2-2.
283H	643	IA32_MC3_CTL2	Module	See Table 2-2.
284H	644	IA32_MC4_CTL2	Package	See Table 2-2.
285H	645	IA32_MC5_CTL2	Package	See Table 2-2.
286H	646	IA32_MC6_CTL2	Package	See Table 2-2.
300H	768	MSR_SGXOWNEREPOCHO	Package	Lower 64 Bit CR_SGXOWNEREPOCH (W)
				Writes do not update CR_SGXOWNEREPOCH if CPUID.(EAX=12H, ECX=0):EAX.SGX1 is 1 on any thread in the package.
		63:0		Lower 64 bits of an 128-bit external entropy value for key derivation of an enclave.
301H	769	MSR_SGXOWNEREPOCH1	Package	Upper 64 Bit CR_SGXOWNEREPOCH (W)
				Writes do not update CR_SGXOWNEREPOCH if CPUID.(EAX=12H, ECX=0):EAX.SGX1 is 1 on any thread in the package.
		63:0		Upper 64 bits of an 128-bit external entropy value for key derivation of an enclave.
38EH	910	IA32_PERF_GLOBAL_STATUS	Соге	See Table 2-2. See Section 20.2.4, "Architectural Performance Monitoring Version 4."
		0		Ovf_PMC0

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		1		Ovf_PMC1
		2		Ovf_PMC2
		3		Ovf_PMC3
		31:4		Reserved
		32		Ovf_FixedCtr0
		33		Ovf_FixedCtr1
		34		Ovf_FixedCtr2
		54:35		Reserved
		55		Trace_ToPA_PMI
		57:56		Reserved
		58		LBR_Frz.
		59		CTR_Frz.
		60		ASCI
		61		Ovf_Uncore
		62		Ovf_BufDSSAVE
		63		CondChgd
390H	912	IA32_PERF_GLOBAL_STATUS_RESET	Core	See Table 2-2. See Section 20.2.4, "Architectural Performance Monitoring Version 4."
		0		Set 1 to clear Ovf_PMCO.
		1		Set 1 to clear Ovf_PMC1.
		2		Set 1 to clear Ovf_PMC2.
		3		Set 1 to clear Ovf_PMC3.
		31:4		Reserved
		32		Set 1 to clear Ovf_FixedCtr0.
		33		Set 1 to clear Ovf_FixedCtr1.
		34		Set 1 to clear Ovf_FixedCtr2.
		54:35		Reserved
		55		Set 1 to clear Trace_ToPA_PMI.
		57:56		Reserved
		58		Set 1 to clear LBR_Frz.
		59		Set 1 to clear CTR_Frz.
		60		Set 1 to clear ASCI.
		61		Set 1 to clear Ovf_Uncore.
		62		Set 1 to clear Ovf_BufDSSAVE.
		63		Set 1 to clear CondChgd.
391H	913	IA32_PERF_GLOBAL_STATUS_SET	Соге	See Table 2-2. See Section 20.2.4, "Architectural Performance Monitoring Version 4."

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

	ister Iress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		0		Set 1 to cause Ovf_PMC0 = 1.
		1		Set 1 to cause Ovf_PMC1 = 1.
		2		Set 1 to cause Ovf_PMC2 = 1.
		3		Set 1 to cause Ovf_PMC3 = 1.
		31:4		Reserved
		32		Set 1 to cause Ovf_FixedCtr0 = 1.
		33		Set 1 to cause Ovf_FixedCtr1 = 1.
		34		Set 1 to cause Ovf_FixedCtr2 = 1.
		54:35		Reserved
		55		Set 1 to cause Trace_ToPA_PMI = 1.
		57:56		Reserved
		58		Set 1 to cause LBR_Frz = 1.
		59		Set 1 to cause CTR_Frz = 1.
		60		Set 1 to cause ASCI = 1.
		61		Set 1 to cause Ovf_Uncore.
		62		Set 1 to cause Ovf_BufDSSAVE.
		63		Reserved
392H	914	IA32_PERF_GLOBAL_INUSE	Соге	See Table 2-2.
3F1H	1009	IA32_PEBS_ENABLE (MSR_PEBS_ENABLE)	Core	See Table 2-2. See Section 20.6.2.4, "Processor Event Based Sampling (PEBS)."
		0		Enable PEBS trigger and recording for the programmed event (precise or otherwise) on IA32_PMCO. (R/W)
3F8H	1016	MSR_PKG_C3_RESIDENCY	Package	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		63:0		Package C3 Residency Counter (R/O)
				Value since last reset that this package is in processor-specific C3 states. Count at the same frequency as the TSC.
3F9H	1017	MSR_PKG_C6_RESIDENCY	Package	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		63:0		Package C6 Residency Counter (R/O)
				Value since last reset that this package is in processor-specific C6 states. Count at the same frequency as the TSC.
3FCH	1020	MSR_CORE_C3_RESIDENCY	Core	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		63:0		CORE C3 Residency Counter (R/O)
				Value since last reset that this core is in processor- specific C3 states. Count at the same frequency as the TSC.
406H	1030	IA32_MC1_ADDR	Module	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
				The IA32_MC2_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MC2_STATUS register is clear.
				When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
418H	1048	IA32_MC6_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
419H	1049	IA32_MC6_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
41AH	1050	IA32_MC6_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
4C3H	1219	IA32_A_PMC2	Core	See Table 2-2.
4C4H	1220	IA32_A_PMC3	Core	See Table 2-2.
4E0H	1248	MSR_SMM_FEATURE_CONTROL	Package	Enhanced SMM Feature Control (SMM-RW)
				Reports SMM capability Enhancement. Accessible only while in SMM.
		0		Lock (SMM-RWO)
				When set to '1' locks this register from further changes.
		1		Reserved
		2		SMM_Code_Chk_En (SMM-RW)
				This control bit is available only if MSR_SMM_MCA_CAP[58] == 1. When set to '0' (default) none of the logical processors are prevented from executing SMM code outside the ranges defined by the SMRR.
				When set to '1' any logical processor in the package that attempts to execute SMM code not within the ranges defined by the SMRR will assert an unrecoverable MCE.
		63:3		Reserved
4E2H	1250	MSR_SMM_DELAYED	Package	SMM Delayed (SMM-R0)
				Reports the interruptible state of all logical processors in the package. Available only while in SMM and MSR_SMM_MCA_CAP[LONG_FLOW_INDICATION] == 1.

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

	ister Iress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		N-1:0		LOG_PROC_STATE (SMM-RO)
				Each bit represents a processor core of its state in a long flow of internal operation which delays servicing an interrupt. The corresponding bit will be set at the start of long events such as: Microcode Update Load, C6, WBINVD, Ratio Change, Throttle.
				The bit is automatically cleared at the end of each long event. The reset value of this field is 0.
				Only bit positions below N = CPUID.(EAX=0BH, ECX=PKG_LVL):EBX[15:0] can be updated.
		63:N		Reserved
4E3H	1251	MSR_SMM_BLOCKED	Package	SMM Blocked (SMM-RO)
				Reports the blocked state of all logical processors in the package. Available only while in SMM.
		N-1:0		LOG_PROC_STATE (SMM-RO)
				Each bit represents a processor core of its blocked state to service an SMI. The corresponding bit will be set if the logical processor is in one of the following states: Wait For SIPI or SENTER Sleep.
				The reset value of this field is OFFFH.
				Only bit positions below N = CPUID.(EAX=0BH, ECX=PKG_LVL):EBX[15:0] can be updated.
		63:N		Reserved
500H	1280	IA32_SGX_SVN_STATUS	Core	Status and SVN Threshold of SGX Support for ACM (R/O)
		0		Lock
				See Section 38.11.3, "Interactions with Authenticated Code Modules (ACMs)".
		15:1		Reserved
		23:16		SGX_SVN_SINIT
				See Section 38.11.3, "Interactions with Authenticated Code Modules (ACMs)".
		63:24		Reserved
560H	1376	IA32_RTIT_OUTPUT_BASE	Соге	Trace Output Base Register (R/W)
				See Table 2-2.
561H	1377	IA32_RTIT_OUTPUT_MASK_PTRS	Core	Trace Output Mask Pointers Register (R/W) See Table 2-2.
570H	1392	IA32_RTIT_CTL	Соге	Trace Control Register (R/W)
		0		TraceEn
		1		CYCEn
		2		OS
		3		User
		6:4		Reserved, must be zero.

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

	ister Iress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		7		CR3Filter
		8		ToPA
				Writing 0 will #GP if also setting TraceEn.
		9		MTCEn
		10		TSCEn
		11		DisRETC
		12		Reserved, must be zero.
		13		BranchEn
		17:14		MTCFreq
		18		Reserved, must be zero.
		22:19		CycThresh
		23		Reserved, must be zero.
		27:24		PSBFreq
		31:28		Reserved, must be zero.
		35:32		ADDRO_CFG
		39:36		ADDR1_CFG
		63:40		Reserved, must be zero.
571H	1393	IA32_RTIT_STATUS	Core	Tracing Status Register (R/W)
		0		FilterEn
				Writes ignored.
		1		ContexEn
				Writes ignored.
		2		TriggerEn
		_		Writes ignored.
		3		Reserved
		4		Error (R/W)
		5		Stopped
		31:6		Reserved, must be zero.
		48:32		PacketByteCnt
		63:49		Reserved, must be zero.
572H	1394	IA32_RTIT_CR3_MATCH	Core	Trace Filter CR3 Match Register (R/W)
		4:0		Reserved
		63:5		CR3[63:5] value to match.
580H	1408	IA32_RTIT_ADDRO_A	Core	Region 0 Start Address (R/W)
		63:0		See Table 2-2.
581H	1409	IA32_RTIT_ADDRO_B	Соге	Region 0 End Address (R/W)
		63:0		See Table 2-2.

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
582H	1410	IA32_RTIT_ADDR1_A	Соге	Region 1 Start Address (R/W)
		63:0		See Table 2-2.
583H	1411	IA32_RTIT_ADDR1_B	Соге	Region 1 End Address (R/W)
		63:0		See Table 2-2.
606H	1542	MSR_RAPL_POWER_UNIT	Package	Unit Multipliers used in RAPL Interfaces (R/O) See Section 15.10.1, "RAPL Interfaces."
		3:0		Power Units
				Power related information (in Watts) is in unit of 1W/2^PU; where PU is an unsigned integer represented by bits 3:0. Default value is 1000b, indicating power unit is in 3.9 milliWatts increment.
		7:4		Reserved
		12:8		Energy Status Units
				Energy related information (in Joules) is in unit of 1 Joule/ (2^ESU); where ESU is an unsigned integer represented by bits 12:8. Default value is 01110b, indicating energy unit is in 61 microJoules.
		15:13		Reserved
		19:16		Time Unit
				Time related information (in seconds) is in unit of 1S/2^TU; where TU is an unsigned integer represented by bits 19:16. Default value is 1010b, indicating power unit is in 0.977 millisecond.
		63:20		Reserved
60AH	1546	MSR_PKGC3_IRTL	Package	Package C3 Interrupt Response Limit (R/W)
				Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		9:0		Interrupt Response Time Limit (R/W)
				Specifies the limit that should be used to decide if the package should be put into a package C3 state.
		12:10		Time Unit (R/W)
				Specifies the encoding value of time unit of the interrupt response time limit. See Table 2-20 for supported time unit encodings.
		14:13		Reserved
		15		Valid (R/W)
				Indicates whether the values in bits 12:0 are valid and can be used by the processor for package C-sate management.
		63:16		Reserved

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
60BH	1547	MSR_PKGC_IRTL1	Package	Package C6/C7S Interrupt Response Limit 1 (R/W) This MSR defines the interrupt response time limit used by the processor to manage a transition to a package C6 or C7S state. Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-states.
		9:0		Interrupt Response Time Limit (R/W)
				Specifies the limit that should be used to decide if the package should be put into a package C6 or C7S state.
		12:10		Time Unit (R/W)
				Specifies the encoding value of time unit of the interrupt response time limit. See Table 2-20 for supported time unit encodings.
		14:13		Reserved
		15		Valid (R/W)
				Indicates whether the values in bits 12:0 are valid and can be used by the processor for package C-sate management.
		63:16		Reserved
60CH	1548	MSR_PKGC_IRTL2	Package	Package C7 Interrupt Response Limit 2 (R/W) This MSR defines the interrupt response time limit used by the processor to manage a transition to a package C7 state. Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		9:0		Interrupt Response Time Limit (R/W)
				Specifies the limit that should be used to decide if the package should be put into a package C7 state.
		12:10		Time Unit (R/W)
				Specifies the encoding value of time unit of the interrupt response time limit. See Table 2-20 for supported time unit encodings.
		14:13		Reserved
		15		Valid (R/W)
				Indicates whether the values in bits 12:0 are valid and can be used by the processor for package C-sate management.
		63:16		Reserved
60DH	1549	MSR_PKG_C2_RESIDENCY	Package	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-states.

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		63:0		Package C2 Residency Counter (R/O) Value since last reset that this package is in processor-specific C2 states. Count at the same frequency as the TSC.
610H	1552	MSR_PKG_POWER_LIMIT	Package	PKG RAPL Power Limit Control (R/W) See Section 15.10.3, "Package RAPL Domain."
611H	1553	MSR_PKG_ENERGY_STATUS	Package	PKG Energy Status (R/O) See Section 15.10.3, "Package RAPL Domain."
613H	1555	MSR_PKG_PERF_STATUS	Package	PKG Perf Status (R/O) See Section 15.10.3, "Package RAPL Domain."
614H	1556	MSR_PKG_POWER_INFO	Package	PKG RAPL Parameters (R/W)
		14:0		Thermal Spec Power (R/W) See Section 15.10.3, "Package RAPL Domain."
		15		Reserved
		30:16		Minimum Power (R/W) See Section 15.10.3, "Package RAPL Domain."
		31		Reserved
		46:32		Maximum Power (R/W) See Section 15.10.3, "Package RAPL Domain."
		47		Reserved
		54:48		Maximum Time Window (R/W) Specified by 2^Y * (1.0 + Z/4.0) * Time_Unit, where "Y" is the unsigned integer value represented by bits 52:48, "Z" is an unsigned integer represented by bits 54:53. "Time_Unit" is specified by the "Time Units" field of MSR_RAPL_POWER_UNIT.
		63:55		Reserved
618H	1560	MSR_DRAM_POWER_LIMIT	Package	DRAM RAPL Power Limit Control (R/W) See Section 15.10.5, "DRAM RAPL Domain."
619H	1561	MSR_DRAM_ENERGY_STATUS	Package	DRAM Energy Status (R/O) See Section 15.10.5, "DRAM RAPL Domain."
61BH	1563	MSR_DRAM_PERF_STATUS	Package	DRAM Performance Throttling Status (R/O) See Section 15.10.5, "DRAM RAPL Domain."
61CH	1564	MSR_DRAM_POWER_INFO	Package	DRAM RAPL Parameters (R/W) See Section 15.10.5, "DRAM RAPL Domain."
632H	1586	MSR_PKG_C10_RESIDENCY	Package	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-states.
		63:0		Package C10 Residency Counter (R/0) Value since last reset that the entire SOC is in an SOi3 state. Count at the same frequency as the TSC.

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
639H	1593	MSR_PPO_ENERGY_STATUS	Package	PPO Energy Status (R/O)
				See Section 15.10.4, "PPO/PP1 RAPL Domains."
641H	1601	MSR_PP1_ENERGY_STATUS	Package	PP1 Energy Status (R/O)
				See Section 15.10.4, "PPO/PP1 RAPL Domains."
64CH	1612	MSR_TURBO_ACTIVATION_RATIO	Package	ConfigTDP Control (R/W)
		7:0		MAX_NON_TURBO_RATIO (RW/L)
				System BIOS can program this field.
		30:8		Reserved
		31		TURBO_ACTIVATION_RATIO_Lock (RW/L)
				When this bit is set, the content of this register is locked until a reset.
		63:32		Reserved
64FH	1615	MSR_CORE_PERF_LIMIT_REASONS	Package	Indicator of Frequency Clipping in Processor Cores (R/W)
				(Frequency refers to processor core frequency.)
		0		PROCHOT Status (R0)
				When set, processor core frequency is reduced below the operating system request due to assertion of external PROCHOT.
		1		Thermal Status (R0)
				When set, frequency is reduced below the operating system request due to a thermal event.
		2		Package-Level Power Limiting PL1 Status (R0)
				When set, frequency is reduced below the operating system request due to package-level power limiting PL1.
		3		Package-Level PL2 Power Limiting Status (R0)
				When set, frequency is reduced below the operating system request due to package-level power limiting PL2.
		8:4		Reserved
		9		Core Power Limiting Status (RO)
				When set, frequency is reduced below the operating system request due to domain-level power limiting.
		10		VR Therm Alert Status (R0)
				When set, frequency is reduced below the operating system request due to a thermal alert from the Voltage Regulator.
		11		Max Turbo Limit Status (R0)
				When set, frequency is reduced below the operating system request due to multi-core turbo limits.

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

Regi Addı		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		12		Electrical Design Point Status (R0) When set, frequency is reduced below the operating system request due to electrical design point constraints (e.g., maximum electrical current consumption).
		13		Turbo Transition Attenuation Status (R0) When set, frequency is reduced below the operating system request due to Turbo transition attenuation. This prevents performance degradation due to frequent operating ratio changes.
		14		Maximum Efficiency Frequency Status (R0) When set, frequency is reduced below the maximum efficiency frequency.
		15		Reserved
		16		PROCHOT Log When set, indicates that the PROCHOT Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.
		17		Thermal Log When set, indicates that the Thermal Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.
		18		Package-Level PL1 Power Limiting Log When set, indicates that the Package Level PL1 Power Limiting Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.
		19		Package-Level PL2 Power Limiting Log When set, indicates that the Package Level PL2 Power Limiting Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.
		24:20		Reserved
		25		Core Power Limiting Log When set, indicates that the Core Power Limiting Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		26		VR Therm Alert Log When set, indicates that the VR Therm Alert Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.
		27		Max Turbo Limit Log When set, indicates that the Max Turbo Limit Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.
		28		Electrical Design Point Log When set, indicates that the EDP Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.
		29		Turbo Transition Attenuation Log When set, indicates that the Turbo Transition Attenuation Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		30		Maximum Efficiency Frequency Log When set, indicates that the Maximum Efficiency Frequency Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software
				writing 0.
		63:31		Reserved
680H	1664	MSR_LASTBRANCH_0_FROM_IP	Core	Last Branch Record O From IP (R/W) One of 32 pairs of last branch record registers on the last branch record stack. The From_IP part of the stack contains pointers to the source instruction . See also: Last Branch Record Stack TOS at 1C9H.
				• Section 18.6 and record format in Section 18.4.8.1.
		0:47		From Linear Address (R/W)
		62:48		Signed extension of bits 47:0.
		63		Mispred
681H	1665	MSR_LASTBRANCH_1_FROM_IP	Core	Last Branch Record 1 From IP (R/W) See description of MSR_LASTBRANCH_O_FROM_IP.
682H	1666	MSR_LASTBRANCH_2_FROM_IP	Core	Last Branch Record 2 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
683H	1667	MSR_LASTBRANCH_3_FROM_IP	Core	Last Branch Record 3 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
684H	1668	MSR_LASTBRANCH_4_FROM_IP	Core	Last Branch Record 4 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
685H	1669	MSR_LASTBRANCH_5_FROM_IP	Core	Last Branch Record 5 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
686H	1670	MSR_LASTBRANCH_6_FROM_IP	Core	Last Branch Record 6 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
687H	1671	MSR_LASTBRANCH_7_FROM_IP	Core	Last Branch Record 7 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
688H	1672	MSR_LASTBRANCH_8_FROM_IP	Core	Last Branch Record 8 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
689H	1673	MSR_LASTBRANCH_9_FROM_IP	Core	Last Branch Record 9 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
68AH	1674	MSR_LASTBRANCH_10_FROM_IP	Core	Last Branch Record 10 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
68BH	1675	MSR_LASTBRANCH_11_FROM_IP	Core	Last Branch Record 11 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
68CH	1676	MSR_LASTBRANCH_12_FROM_IP	Core	Last Branch Record 12 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
68DH	1677	MSR_LASTBRANCH_13_FROM_IP	Core	Last Branch Record 13 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
68EH	1678	MSR_LASTBRANCH_14_FROM_IP	Core	Last Branch Record 14 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
68FH	1679	MSR_LASTBRANCH_15_FROM_IP	Core	Last Branch Record 15 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
690H	1680	MSR_LASTBRANCH_16_FROM_IP	Core	Last Branch Record 16 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
691H	1681	MSR_LASTBRANCH_17_FROM_IP	Соге	Last Branch Record 17 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
692H	1682	MSR_LASTBRANCH_18_FROM_IP	Core	Last Branch Record 18 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
693H	1683	MSR_LASTBRANCH_19_FROM_IP	Соге	Last Branch Record 19From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
694H	1684	MSR_LASTBRANCH_20_FROM_IP	Core	Last Branch Record 20 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
695H	1685	MSR_LASTBRANCH_21_FROM_IP	Core	Last Branch Record 21 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
696H	1686	MSR_LASTBRANCH_22_FROM_IP	Core	Last Branch Record 22 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
697H	1687	MSR_LASTBRANCH_23_FROM_IP	Core	Last Branch Record 23 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
698H	1688	MSR_LASTBRANCH_24_FROM_IP	Core	Last Branch Record 24 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
699H	1689	MSR_LASTBRANCH_25_FROM_IP	Core	Last Branch Record 25 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
69AH	1690	MSR_LASTBRANCH_26_FROM_IP	Core	Last Branch Record 26 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
69BH	1691	MSR_LASTBRANCH_27_FROM_IP	Core	Last Branch Record 27 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
69CH	1692	MSR_LASTBRANCH_28_FROM_IP	Core	Last Branch Record 28 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
69DH	1693	MSR_LASTBRANCH_29_FROM_IP	Core	Last Branch Record 29 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
69EH	1694	MSR_LASTBRANCH_30_FROM_IP	Core	Last Branch Record 30 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
69FH	1695	MSR_LASTBRANCH_31_FROM_IP	Соге	Last Branch Record 31 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
6COH	1728	MSR_LASTBRANCH_0_TO_IP	Core	Last Branch Record 0 To IP (R/W) One of 32 pairs of last branch record registers on the last branch record stack. The To_IP part of the stack contains pointers to the Destination instruction and elapsed cycles from last LBR update. See Section 18.6.
		0:47		Target Linear Address (R/W)
		63:48		Elapsed cycles from last update to the LBR.
6C1H	1729	MSR_LASTBRANCH_1_TO_IP	Core	Last Branch Record 1 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6C2H	1730	MSR_LASTBRANCH_2_TO_IP	Core	Last Branch Record 2 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6C3H	1731	MSR_LASTBRANCH_3_TO_IP	Core	Last Branch Record 3 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6C4H	1732	MSR_LASTBRANCH_4_TO_IP	Core	Last Branch Record 4 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6C5H	1733	MSR_LASTBRANCH_5_TO_IP	Core	Last Branch Record 5 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6C6H	1734	MSR_LASTBRANCH_6_TO_IP	Core	Last Branch Record 6 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6C7H	1735	MSR_LASTBRANCH_7_TO_IP	Core	Last Branch Record 7 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
6C8H	1736	MSR_LASTBRANCH_8_TO_IP	Соге	Last Branch Record 8 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6C9H	1737	MSR_LASTBRANCH_9_TO_IP	Core	Last Branch Record 9 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6CAH	1738	MSR_LASTBRANCH_10_T0_IP	Core	Last Branch Record 10 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6CBH	1739	MSR_LASTBRANCH_11_TO_IP	Соге	Last Branch Record 11 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6CCH	1740	MSR_LASTBRANCH_12_TO_IP	Core	Last Branch Record 12 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6CDH	1741	MSR_LASTBRANCH_13_TO_IP	Core	Last Branch Record 13 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6CEH	1742	MSR_LASTBRANCH_14_TO_IP	Core	Last Branch Record 14 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6CFH	1743	MSR_LASTBRANCH_15_TO_IP	Core	Last Branch Record 15 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6D0H	1744	MSR_LASTBRANCH_16_TO_IP	Соге	Last Branch Record 16 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6D1H	1745	MSR_LASTBRANCH_17_TO_IP	Core	Last Branch Record 17 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6D2H	1746	MSR_LASTBRANCH_18_TO_IP	Core	Last Branch Record 18 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6D3H	1747	MSR_LASTBRANCH_19_TO_IP	Core	Last Branch Record 19To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6D4H	1748	MSR_LASTBRANCH_20_TO_IP	Core	Last Branch Record 20 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6D5H	1749	MSR_LASTBRANCH_21_TO_IP	Core	Last Branch Record 21 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6D6H	1750	MSR_LASTBRANCH_22_TO_IP	Core	Last Branch Record 22 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6D7H	1751	MSR_LASTBRANCH_23_TO_IP	Core	Last Branch Record 23 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6D8H	1752	MSR_LASTBRANCH_24_TO_IP	Core	Last Branch Record 24 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6D9H	1753	MSR_LASTBRANCH_25_TO_IP	Core	Last Branch Record 25 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6DAH	1754	MSR_LASTBRANCH_26_TO_IP	Core	Last Branch Record 26 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
6DBH	1755	MSR_LASTBRANCH_27_TO_IP	Core	Last Branch Record 27 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6DCH	1756	MSR_LASTBRANCH_28_TO_IP	Core	Last Branch Record 28 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6DDH	1757	MSR_LASTBRANCH_29_TO_IP	Core	Last Branch Record 29 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6DEH	1758	MSR_LASTBRANCH_30_TO_IP	Core	Last Branch Record 30 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6DFH	1759	MSR_LASTBRANCH_31_TO_IP	Core	Last Branch Record 31 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
802H	2050	IA32_X2APIC_APICID	Соге	x2APIC ID register (R/O)
803H	2051	IA32_X2APIC_VERSION	Core	x2APIC Version register (R/O)
808H	2056	IA32_X2APIC_TPR	Core	x2APIC Task Priority register (R/W)
80AH	2058	IA32_X2APIC_PPR	Core	x2APIC Processor Priority register (R/O)
80BH	2059	IA32_X2APIC_EOI	Core	x2APIC EOI register (W/O)
80DH	2061	IA32_X2APIC_LDR	Core	x2APIC Logical Destination register (R/O)
80FH	2063	IA32_X2APIC_SIVR	Core	x2APIC Spurious Interrupt Vector register (R/W)
810H	2064	IA32_X2APIC_ISR0	Core	x2APIC In-Service register bits [31:0] (R/0)
811H	2065	IA32_X2APIC_ISR1	Соге	x2APIC In-Service register bits [63:32] (R/0)
812H	2066	IA32_X2APIC_ISR2	Core	x2APIC In-Service register bits [95:64] (R/0)
813H	2067	IA32_X2APIC_ISR3	Соге	x2APIC In-Service register bits [127:96] (R/0)
814H	2068	IA32_X2APIC_ISR4	Соге	x2APIC In-Service register bits [159:128] (R/0)
815H	2069	IA32_X2APIC_ISR5	Core	x2APIC In-Service register bits [191:160] (R/0)
816H	2070	IA32_X2APIC_ISR6	Соге	x2APIC In-Service register bits [223:192] (R/0)
817H	2071	IA32_X2APIC_ISR7	Соге	x2APIC In-Service register bits [255:224] (R/0)
818H	2072	IA32_X2APIC_TMR0	Соге	x2APIC Trigger Mode register bits [31:0] (R/0)
819H	2073	IA32_X2APIC_TMR1	Соге	x2APIC Trigger Mode register bits [63:32] (R/O)
81AH	2074	IA32_X2APIC_TMR2	Соге	x2APIC Trigger Mode register bits [95:64] (R/O)
81BH	2075	IA32_X2APIC_TMR3	Core	x2APIC Trigger Mode register bits [127:96] (R/O)
81CH	2076	IA32_X2APIC_TMR4	Core	x2APIC Trigger Mode register bits [159:128] (R/0)
81DH	2077	IA32_X2APIC_TMR5	Соге	x2APIC Trigger Mode register bits [191:160] (R/0)
81EH	2078	IA32_X2APIC_TMR6	Core	x2APIC Trigger Mode register bits [223:192] (R/0)
81FH	2079	IA32_X2APIC_TMR7	Core	x2APIC Trigger Mode register bits [255:224] (R/0)
820H	2080	IA32_X2APIC_IRR0	Core	x2APIC Interrupt Request register bits [31:0] (R/O)
821H	2081	IA32_X2APIC_IRR1	Core	x2APIC Interrupt Request register bits [63:32] (R/O)
822H	2082	IA32_X2APIC_IRR2	Core	x2APIC Interrupt Request register bits [95:64] (R/O)
823H	2083	IA32_X2APIC_IRR3	Соге	x2APIC Interrupt Request register bits [127:96] (R/O)

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
824H	2084	IA32_X2APIC_IRR4	Core	x2APIC Interrupt Request register bits [159:128] (R/O)
825H	2085	IA32_X2APIC_IRR5	Core	x2APIC Interrupt Request register bits [191:160] (R/O)
826H	2086	IA32_X2APIC_IRR6	Core	x2APIC Interrupt Request register bits [223:192] (R/O)
827H	2087	IA32_X2APIC_IRR7	Core	x2APIC Interrupt Request register bits [255:224] (R/O)
828H	2088	IA32_X2APIC_ESR	Соге	x2APIC Error Status register (R/W)
82FH	2095	IA32_X2APIC_LVT_CMCI	Core	x2APIC LVT Corrected Machine Check Interrupt register (R/W)
830H	2096	IA32_X2APIC_ICR	Соге	x2APIC Interrupt Command register (R/W)
832H	2098	IA32_X2APIC_LVT_TIMER	Соге	x2APIC LVT Timer Interrupt register (R/W)
833H	2099	IA32_X2APIC_LVT_THERMAL	Соге	x2APIC LVT Thermal Sensor Interrupt register (R/W)
834H	2100	IA32_X2APIC_LVT_PMI	Соге	x2APIC LVT Performance Monitor register (R/W)
835H	2101	IA32_X2APIC_LVT_LINTO	Соге	x2APIC LVT LINTO register (R/W)
836H	2102	IA32_X2APIC_LVT_LINT1	Соге	x2APIC LVT LINT1 register (R/W)
837H	2103	IA32_X2APIC_LVT_ERROR	Соге	x2APIC LVT Error register (R/W)
838H	2104	IA32_X2APIC_INIT_COUNT	Соге	x2APIC Initial Count register (R/W)
839H	2105	IA32_X2APIC_CUR_COUNT	Соге	x2APIC Current Count register (R/O)
83EH	2110	IA32_X2APIC_DIV_CONF	Соге	x2APIC Divide Configuration register (R/W)
83FH	2111	IA32_X2APIC_SELF_IPI	Соге	x2APIC Self IPI register (W/O)
C8FH	3215	IA32_PQR_ASSOC	Соге	Resource Association Register (R/W)
		31:0		Reserved
		33:32		COS (R/W)
		63: 34		Reserved
D10H	3344	IA32_L2_QOS_MASK_0	Module	L2 Class Of Service Mask - COS O (R/W)
				If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=0.
		0:7		CBM: Bit vector of available L2 ways for COS 0 enforcement.
		63:8		Reserved
D11H	3345	IA32_L2_QOS_MASK_1	Module	L2 Class Of Service Mask - COS 1 (R/W)
				If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=1.
		0:7		CBM: Bit vector of available L2 ways for COS 0 enforcement.
		63:8		Reserved
D12H	3346	IA32_L2_QOS_MASK_2	Module	L2 Class Of Service Mask - COS 2 (R/W)
				If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=2.
		0:7		CBM: Bit vector of available L2 ways for COS 0 enforcement.

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		63:8		Reserved
D13H	3347	IA32_L2_QOS_MASK_3	Package	L2 Class Of Service Mask - COS 3 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=3.
		0:19		CBM: Bit vector of available L2 ways for COS 3 enforcement.
		63:20		Reserved
D90H	3472	IA32_BNDCFGS	Core	See Table 2-2.
DAOH	3488	IA32_XSS	Core	See Table 2-2.
See Tabl	e 2-6, and	Table 2-12 for MSR definitions applicable	to processors w	ith CPUID signature 06_5CH.

Table 2-12. MSRs in Intel Atom® Processors Based on the Goldmont Microarchitecture (Contd.)

2.6 MSRS IN INTEL ATOM® PROCESSORS BASED ON GOLDMONT PLUS MICROARCHITECTURE

Intel Atom processors based on the Goldmont Plus microarchitecture support MSRs listed in Table 2-6, Table 2-12, and Table 2-13. These processors have a CPUID signature with DisplayFamily_DisplayModel of 06_7AH; see Table 2-1. For an MSR listed in Table 2-13 that also appears in the model-specific tables of prior generations, Table 2-13 supersede prior generation tables.

In the Goldmont Plus microarchitecture, the scope column indicates the following: "Core" means each processor core has a separate MSR, or a bit field not shared with another processor core. "Module" means the MSR or the bit field is shared by a subset of the processor cores in the physical package. The number of processor cores in this subset is model specific and may differ between different processors. For all processors based on Goldmont Plus microarchitecture, the L2 cache is also shared between cores in a module and thus CPUID leaf 04H enumeration can be used to figure out which processors are in the same module. "Package" means all processor cores in the physical package share the same MSR or bit interface.

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
ЗАН	58	IA32_FEATURE_CONTROL	Core	Control Features in Intel 64Processor (R/W)
				See Table 2-2.
		0		Lock (R/WL)
		1		Enable VMX inside SMX operation (R/WL)
		2		Enable VMX outside SMX operation (R/WL)
		14:8		SENTER local functions enables (R/WL)
		15		SENTER global functions enable (R/WL)
		17		SGX Launch Control Enable (R/WL)
				This bit must be set to enable runtime reconfiguration of SGX Launch Control via IA32_SGXLEPUBKEYHASHN MSR.
				Valid if CPUID.(EAX=07H, ECX=0H): ECX[30] = 1.
		18		SGX global functions enable (R/WL)

Table 2-13. MSRs in Intel Atom® Processors Based on the Goldmont Plus Microarchitecture

Table 2-13. MSRs in Intel Atom® Processors Based on the Goldmont Plus Microarchitecture (Contd.)

	ister Iress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		63:19		Reserved
8CH	140	IA32_SGXLEPUBKEYHASH0	Core	See Table 2-2.
8DH	141	IA32_SGXLEPUBKEYHASH1	Core	See Table 2-2.
8EH	142	IA32_SGXLEPUBKEYHASH2	Core	See Table 2-2.
8FH	143	IA32_SGXLEPUBKEYHASH3	Core	See Table 2-2.
3F1H	1009	IA32_PEBS_ENABLE (MSR_PEBS_ENABLE)	Core	(R/W) See Table 2-2. See Section 20.6.2.4, "Processor Event Based Sampling (PEBS)."
		0		Enable PEBS trigger and recording for the programmed event (precise or otherwise) on IA32_PMCO.
		1		Enable PEBS trigger and recording for the programmed event (precise or otherwise) on IA32_PMC1.
		2		Enable PEBS trigger and recording for the programmed event (precise or otherwise) on IA32_PMC2.
		3		Enable PEBS trigger and recording for the programmed event (precise or otherwise) on IA32_PMC3.
		31:4		Reserved
		32		Enable PEBS trigger and recording for IA32_FIXED_CTRO.
		33		Enable PEBS trigger and recording for IA32_FIXED_CTR1.
		34		Enable PEBS trigger and recording for IA32_FIXED_CTR2.
		63:35		Reserved
570H	1392	IA32_RTIT_CTL	Core	Trace Control Register (R/W)
		0		TraceEn
		1		CYCEn
		2		OS
		3		User
		4		PwrEvtEn
		5		FUPonPTW
		6		FabricEn
		7		CR3Filter
		8		ТоРА
				Writing 0 will #GP if also setting TraceEn.
		9		MTCEn
		10		TSCEn

Table 2-13. MSRs in Intel Atom® Processors Based on the Goldmont Plus Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		11		DisRETC
		12		PTWEn
		13		BranchEn
		17:14		MTCFreq
		18		Reserved, must be zero.
		22:19		CycThresh
		23		Reserved, must be zero.
		27:24		PSBFreq
		31:28		Reserved, must be zero.
		35:32		ADDRO_CFG
		39:36		ADDR1_CFG
		63:40		Reserved, must be zero.
680H	1664	MSR_LASTBRANCH_0_FROM_IP	Соге	Last Branch Record O From IP (R/W)
				One of the three MSRs that make up the first entry of the 32-entry LBR stack. The From_IP part of the stack contains pointers to the source instruction. See also:
				 Last Branch Record Stack TOS at 1C9H. Section 18.7, "Last Branch, Call Stack, Interrupt, and Exception Recording for Processors based on Goldmont Plus Microarchitecture."
681H	1665	MSR_LASTBRANCH_ <i>i_</i> FROM_IP	Соге	Last Branch Record <i>i</i> From IP (R/W)
- 69FH	- 1695			See description of MSR_LASTBRANCH_0_FROM_IP; <i>i</i> = 1-31.
6C0H	1728	MSR_LASTBRANCH_0_TO_IP	Core	Last Branch Record 0 To IP (R/W)
				One of the three MSRs that make up the first entry of the 32-entry LBR stack. The To_IP part of the stack contains pointers to the Destination instruction. See also:
				 Section 18.7, "Last Branch, Call Stack, Interrupt, and Exception Recording for Processors based on Goldmont Plus Microarchitecture."
6C1H	1729	MSR_LASTBRANCH_i_TO_IP	Core	Last Branch Record i To IP (R/W)
- 6DFH	- 1759			See description of MSR_LASTBRANCH_0_T0_IP; <i>i</i> = 1-31.
DCOH	3520	MSR_LASTBRANCH_INFO_0	Core	Last Branch Record O Additional Information (R/W)
				One of the three MSRs that make up the first entry of the 32-entry LBR stack. This part of the stack contains flag and elapsed cycle information. See also:
				Last Branch Record Stack TOS at 1C9H.Section 18.9.1, "LBR Stack."
DC1H	3521	MSR_LASTBRANCH_INFO_1	Core	Last Branch Record 1 Additional Information (R/W)
				See description of MSR_LASTBRANCH_INFO_0.

Table 2-13. MSRs in Intel Atom® Processors Based on the Goldmont Plus Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
DC2H	3522	MSR_LASTBRANCH_INFO_2	Core	Last Branch Record 2 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DC3H	3523	MSR_LASTBRANCH_INFO_3	Core	Last Branch Record 3 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DC4H	3524	MSR_LASTBRANCH_INFO_4	Core	Last Branch Record 4 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DC5H	3525	MSR_LASTBRANCH_INFO_5	Core	Last Branch Record 5 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DC6H	3526	MSR_LASTBRANCH_INFO_6	Core	Last Branch Record 6 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DC7H	3527	MSR_LASTBRANCH_INFO_7	Core	Last Branch Record 7 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DC8H	3528	MSR_LASTBRANCH_INFO_8	Core	Last Branch Record 8 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DC9H	3529	MSR_LASTBRANCH_INFO_9	Core	Last Branch Record 9 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DCAH	3530	MSR_LASTBRANCH_INFO_10	Core	Last Branch Record 10 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DCBH	3531	MSR_LASTBRANCH_INFO_11	Core	Last Branch Record 11 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DCCH	3532	MSR_LASTBRANCH_INFO_12	Core	Last Branch Record 12 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DCDH	3533	MSR_LASTBRANCH_INFO_13	Core	Last Branch Record 13 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DCEH	3534	MSR_LASTBRANCH_INFO_14	Core	Last Branch Record 14 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DCFH	3535	MSR_LASTBRANCH_INFO_15	Core	Last Branch Record 15 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DD0H	3536	MSR_LASTBRANCH_INFO_16	Core	Last Branch Record 16 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DD1H	3537	MSR_LASTBRANCH_INFO_17	Core	Last Branch Record 17 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DD2H	3538	MSR_LASTBRANCH_INFO_18	Core	Last Branch Record 18 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DD3H	3539	MSR_LASTBRANCH_INFO_19	Core	Last Branch Record 19 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DD4H	3520	MSR_LASTBRANCH_INFO_20	Core	Last Branch Record 20 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.

Table 2-13. MSRs in Intel Atom® Processors Based on the Goldmont Plus Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
DD5H	3521	MSR_LASTBRANCH_INFO_21	Core	Last Branch Record 21 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DD6H	3522	MSR_LASTBRANCH_INFO_22	Core	Last Branch Record 22 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DD7H	3523	MSR_LASTBRANCH_INFO_23	Core	Last Branch Record 23 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DD8H	3524	MSR_LASTBRANCH_INFO_24	Core	Last Branch Record 24 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DD9H	3525	MSR_LASTBRANCH_INFO_25	Core	Last Branch Record 25 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DDAH	3526	MSR_LASTBRANCH_INFO_26	Core	Last Branch Record 26 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DDBH	3527	MSR_LASTBRANCH_INFO_27	Core	Last Branch Record 27 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DDCH	3528	MSR_LASTBRANCH_INFO_28	Core	Last Branch Record 28 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DDDH	3529	MSR_LASTBRANCH_INFO_29	Core	Last Branch Record 29 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DDEH	3530	MSR_LASTBRANCH_INFO_30	Core	Last Branch Record 30 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.
DDFH	3531	MSR_LASTBRANCH_INFO_31	Core	Last Branch Record 31 Additional Information (R/W) See description of MSR_LASTBRANCH_INFO_0.

2.7 MSRS IN INTEL ATOM® PROCESSORS BASED ON TREMONT MICROARCHITECTURE

Processors based on the Tremont microarchitecture support MSRs listed in Table 2-6, Table 2-12, Table 2-13, and Table 2-14. These processors have a CPUID signature with DisplayFamily_DisplayModel of 06_86H, 06_96H, or 06_9CH; see Table 2-1. For an MSR listed in Table 2-14 that also appears in the model-specific tables of prior generations, Table 2-14 supersede prior generation tables.

In the Tremont microarchitecture, the scope column indicates the following: "Core" means each processor core has a separate MSR, or a bit field not shared with another processor core. "Module" means the MSR or the bit field is shared by a subset of the processor cores in the physical package. The number of processor cores in this subset is model specific and may differ between different processors. For all processors based on Tremont microarchitecture, the L2 cache is also shared between cores in a module and thus CPUID leaf 04H enumeration can be used to figure out which processors are in the same module. "Package" means all processor cores in the physical package share the same MSR or bit interface.

Table 2-14. MSRs in Intel Atom® Processors Based on the Tremont Microarchitecture

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
33H	51	MSR_MEMORY_CTRL	Соге	Memory Control Register
		28:0		Reserved.
		29		Enable #AC(0) exception for split locked accesses:
				Cause #AC(0) exception for split locked access at all CPL irrespective of CRO.AM or EFLAGS.AC.
		30		Reserved.
		31		Reserved.
CFH	207	IA32_CORE_CAPABILITIES	Core	IA32 Core Capabilities Register If CPUID.(EAX=07H, ECX=0):EDX[30] = 1.
		4:0		Reserved.
		5		Bit 29 of MSR_MEMORY_CTRL (address 33H) supported.
		63:6		Reserved.
2A0H	672	MSR_PRMRR_BASE_0	Соге	Processor Reserved Memory Range Register - Physical Base Control Register (R/W)
		2:0		MEMTYPE: PRMRR BASE Memory Type.
		3		CONFIGURED: PRMRR BASE Configured.
		11:4		Reserved.
		51:12		BASE: PRMRR Base Address.
		63:52		Reserved.
3F1H	1009	IA32_PEBS_ENABLE (MSR_PEBS_ENABLE)	Core	(R/W) See Table 2-2. See Section 20.6.2.4, "Processor Event Based Sampling (PEBS)".
		<i>n</i> :0		Enable PEBS trigger and recording for the programmed event (precise or otherwise) on IA32_PMCx. The maximum value n can be determined from CPUID.0AH:EAX[15:8].
		31:n+1		Reserved.
		32+ <i>m</i> :32		Enable PEBS trigger and recording for IA32_FIXED_CTRx. The maximum value m can be determined from CPUID.0AH:EDX[4:0].
		59:33+m		Reserved.
		60		Pend a PerfMon Interrupt (PMI) after each PEBS event.
		62:61		Specifies PEBS output destination. Encodings:
				00B: DS Save Area
				01B: Intel PT trace output. Supported if IA32_PERF_CAPABILITIES.PEBS_OUTPUT_PT_AVAIL[16] and CPUID.07H.0.EBX[25] are set.
				10B: Reserved
				11B: Reserved
		63		Reserved.

Table 2-14. MSRs in Intel Atom® Processors Based on the Tremont Microarchitecture (Contd.)

Regi Addı		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
1309H	4873	MSR_RELOAD_FIXED_CTRx		Reload value for IA32_FIXED_CTRx (R/W)
- 130BH	- 4875	47:0		Value loaded into IA32_FIXED_CTRx when a PEBS record is generated while PEBS_EN_FIXEDx = 1 and PEBS_OUTPUT = 01B in IA32_PEBS_ENABLE, and FIXED_CTRx is overflowed.
		63:48		Reserved.
14C1H	5313	MSR_RELOAD_PMCx	Core	Reload value for IA32_PMCx (R/W)
- 14C4H	- 5316	47:0		Value loaded into IA32_PMCx when a PEBS record is generated while PEBS_EN_PMCx = 1 and PEBS_OUTPUT = 01B in IA32_PEBS_ENABLE, and PMCx is overflowed.
		63:48		Reserved.
See Table	e 2-6, Tab	lle 2-12, Table 2-13, and Table 2-14 for MS	R definitions app	plicable to processors with CPUID signature 06_86H.

2.8 MSRS IN THE NEHALEM MICROARCHITECTURE

Table 2-15 lists model-specific registers (MSRs) that are common for Nehalem microarchitecture. These include the Intel Core i7 and i5 processor family. These processors have a CPUID signature with DisplayFamily_DisplayModel of 06_1AH, 06_1EH, 06_1FH, 06_2EH, see Table 2-1. Additional MSRs specific to 06_1AH, 06_1EH, 06_1FH are listed in Table 2-16. Some MSRs listed in these tables are used by BIOS. More information about these MSR can be found at http://biosbits.org.

The column "Scope" represents the package/core/thread scope of individual bit field of an MSR. "Thread" means this bit field must be programmed on each logical processor independently. "Core" means the bit field must be programmed on each processor core independently, logical processors in the same core will be affected by change of this bit on the other logical processor in the same core. "Package" means the bit field must be programmed once for each physical package. Change of a bit filed with a package scope will affect all logical processors in that physical package.

Table 2-15. MSRs in Processors Based on Nehalem Microarchitecture

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
OH	0	IA32_P5_MC_ADDR	Thread	See Section 2.23, "MSRs in Pentium Processors."
1H	1	IA32_P5_MC_TYPE	Thread	See Section 2.23, "MSRs in Pentium Processors."
6H	6	IA32_MONITOR_FILTER_SIZE	Thread	See Section 9.10.5, "Monitor/Mwait Address Range Determination" and Table 2-2.
10H	16	IA32_TIME_STAMP_COUNTER	Thread	See Section 18.17, "Time-Stamp Counter," and see Table 2-2.
17H	23	IA32_PLATFORM_ID	Package	Platform ID (R) See Table 2-2.
17H	23	MSR_PLATFORM_ID	Package	Model Specific Platform ID (R)
		49:0		Reserved

Table 2-15. MSRs in Processors Based on Nehalem Microarchitecture (Contd.)

	ister Iress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		52:50		See Table 2-2.
		63:53		Reserved
1BH	27	IA32_APIC_BASE	Thread	See Section 11.4.4, "Local APIC Status and Location," and Table 2-2.
34H	52	MSR_SMI_COUNT	Thread	SMI Counter (R/O)
		31:0		SMI Count (R/O)
		C2:22		Running count of SMI events since last RESET. Reserved
2011	50	63:32		
ЗАН	58	IA32_FEATURE_CONTROL	Thread	Control Features in Intel 64Processor (R/W) See Table 2-2.
79H	121	IA32_BIOS_UPDT_TRIG	Соге	BIOS Update Trigger Register (W)
				See Table 2-2.
8BH	139	IA32_BIOS_SIGN_ID	Thread	BIOS Update Signature ID (R/W) See Table 2-2.
C1H	193	IA32_PMC0	Thread	Performance Counter Register
				See Table 2-2.
C2H	194	IA32_PMC1	Thread	Performance Counter Register
				See Table 2-2.
СЗН	195	IA32_PMC2	Thread	Performance Counter Register
				See Table 2-2.
C4H	196	IA32_PMC3	Thread	Performance Counter Register
				See Table 2-2.
CEH	206	MSR_PLATFORM_INFO	Package	Platform Information
				Contains power management and other model specific features enumeration. See http://biosbits.org.
		7:0		Reserved
		15:8	Package	Maximum Non-Turbo Ratio (R/O)
				This is the ratio of the frequency that invariant TSC runs at. The invariant TSC frequency can be computed by multiplying this ratio by 133.33 MHz.
		27:16		Reserved
		28	Package	Programmable Ratio Limit for Turbo Mode (R/O)
				When set to 1, indicates that Programmable Ratio Limit for Turbo mode is enabled. When set to 0, indicates Programmable Ratio Limit for Turbo mode is disabled.
		29	Package	Programmable TDC-TDP Limit for Turbo Mode (R/O)
				When set to 1, indicates that TDC and TDP Limits for Turbo mode are programmable. When set to 0, indicates TDC and TDP Limits for Turbo mode are not programmable.

Table 2-15. MSRs in Processors Based on Nehalem Microarchitecture (Contd.)

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		39:30		Reserved
		47:40	Package	Maximum Efficiency Ratio (R/O)
				This is the minimum ratio (maximum efficiency) that the processor can operate, in units of 133.33MHz.
		63:48		Reserved
E2H	226	MSR_PKG_CST_CONFIG_CONTROL	Соге	C-State Configuration Control (R/W)
				Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States. See http://biosbits.org.
		2:0		Package C-State Limit (R/W)
				Specifies the lowest processor-specific C-state code name (consuming the least power) for the package. The default is set as factory-configured package C-state limit.
				The following C-state code name encodings are supported:
				000b: CO (no package C-sate support)
				001b: C1 (Behavior is the same as 000b)
				010b: C3
				011b: C6
				100b: C7
				101b and 110b: Reserved
				111: No package C-state limit.
				Note: This field cannot be used to limit package C-state to C3.
		9:3		Reserved
		10		I/O MWAIT Redirection Enable (R/W)
				When set, will map IO_read instructions sent to IO register specified by MSR_PMG_IO_CAPTURE_BASE to MWAIT instructions.
		14:11		Reserved
		15		CFG Lock (R/WO)
				When set, locks bits 15:0 of this register until next reset.
		23:16		Reserved
		24		Interrupt filtering enable (R/W)
				When set, processor cores in a deep C-State will wake only when the event message is destined for that core. When O, all processor cores in a deep C-State will wake for an event message.
		25		C3 state auto demotion enable (R/W)
				When set, the processor will conditionally demote C6/C7 requests to C3 based on uncore auto-demote information.

Table 2-15. MSRs in Processors Based on Nehalem Microarchitecture (Contd.)

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		26		C1 state auto demotion enable (R/W) When set, the processor will conditionally demote C3/C6/C7 requests to C1 based on uncore auto-demote information.
		27		Enable C3 Undemotion (R/W)
		28		Enable C1 Undemotion (R/W)
		29		Package C State Demotion Enable (R/W)
		30		Package C State UnDemotion Enable (R/W)
		63:31		Reserved
E4H	228	MSR_PMG_IO_CAPTURE_BASE	Core	Power Management IO Redirection in C-state (R/W) See http://biosbits.org.
		15:0 18:16		LVL_2 Base Address (R/W) Specifies the base address visible to software for IO redirection. If IO MWAIT Redirection is enabled, reads to this address will be consumed by the power management logic and decoded to MWAIT instructions. When IO port address redirection is enabled, this is the IO port address reported to the OS/software. C-state Range (R/W) Specifies the encoding value of the maximum C-State code name to be included when IO read to MWAIT redirection is enabled by MSR_PKG_CST_CONFIG_CONTROL[bit10]:
				000b - C3 is the max C-State to include. 001b - C6 is the max C-State to include. 010b - C7 is the max C-State to include.
		63:19		Reserved
E7H	231	IA32_MPERF	Thread	Maximum Performance Frequency Clock Count (R/W) See Table 2-2.
E8H	232	IA32_APERF	Thread	Actual Performance Frequency Clock Count (R/W) See Table 2-2.
FEH	254	IA32_MTRRCAP	Thread	See Table 2-2.
174H	372	IA32_SYSENTER_CS	Thread	See Table 2-2.
175H	373	IA32_SYSENTER_ESP	Thread	See Table 2-2.
176H	374	IA32_SYSENTER_EIP	Thread	See Table 2-2.
179H	377	IA32_MCG_CAP	Thread	See Table 2-2.
17AH	378	IA32_MCG_STATUS	Thread	Global Machine Check Status

Table 2-15. MSRs in Processors Based on Nehalem Microarchitecture (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		0		RIPV When set, bit indicates that the instruction addressed by the instruction pointer pushed on the stack (when the machine check was generated) can be used to restart the program. If cleared, the program cannot be reliably restarted.
		1		EIPV
				When set, bit indicates that the instruction addressed by the instruction pointer pushed on the stack (when the machine check was generated) is directly associated with the error.
		2		MCIP
				When set, bit indicates that a machine check has been generated. If a second machine check is detected while this bit is still set, the processor enters a shutdown state. Software should write this bit to 0 after processing a machine check exception.
		63:3		Reserved
186H	390	IA32_PERFEVTSEL0	Thread	See Table 2-2.
		7:0		Event Select
		15:8		UMask
		16		USR
		17		OS
		18		Edge
		19		PC
		20		INT
		21		AnyThread
		22		EN
		23		INV
		31:24		CMASK
		63:32		Reserved
187H	391	IA32_PERFEVTSEL1	Thread	See Table 2-2.
188H	392	IA32_PERFEVTSEL2	Thread	See Table 2-2.
189H	393	IA32_PERFEVTSEL3	Thread	See Table 2-2.
198H	408	IA32_PERF_STATUS	Core	See Table 2-2.
		15:0		Current Performance State Value.
		63:16		Reserved
199H	409	IA32_PERF_CTL	Thread	See Table 2-2.

Table 2-15. MSRs in Processors Based on Nehalem Microarchitecture (Contd.)

Regi Addı		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	1		
19AH	410	IA32_CLOCK_MODULATION	Thread	Clock Modulation (R/W) See Table 2-2. IA32_CLOCK_MODULATION MSR was originally named IA32_THERM_CONTROL MSR.
		0		Reserved
		3:1		On demand Clock Modulation Duty Cycle (R/W)
		4		On demand Clock Modulation Enable (R/W)
		63:5		Reserved
19BH	411	IA32_THERM_INTERRUPT	Core	Thermal Interrupt Control (R/W) See Table 2-2.
19CH	412	IA32_THERM_STATUS	Соге	Thermal Monitor Status (R/W)
				See Table 2-2.
1A0H	416	IA32_MISC_ENABLE		Enable Misc. Processor Features (R/W)
				Allows a variety of processor functions to be enabled and disabled.
		0	Thread	Fast-Strings Enable
				See Table 2-2.
		2:1		Reserved
		3	Thread	Automatic Thermal Control Circuit Enable (R/W)
				See Table 2-2. Default value is 1.
		6:4		Reserved
		7	Thread	Performance Monitoring Available (R) See Table 2-2.
		10:8		Reserved
		11	Thread	Branch Trace Storage Unavailable (R/O) See Table 2-2.
		12	Thread	Processor Event Based Sampling Unavailable (R/O) See Table 2-2.
		15:13		Reserved
		16	Package	Enhanced Intel SpeedStep Technology Enable (R/W) See Table 2-2.
		18	Thread	ENABLE MONITOR FSM. (R/W) See Table 2-2.
		21:19		Reserved
		22	Thread	Limit CPUID Maxval (R/W) See Table 2-2.
		23	Thread	xTPR Message Disable (R/W) See Table 2-2.
		33:24		Reserved

Table 2-15. MSRs in Processors Based on Nehalem Microarchitecture (Contd.)

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		34	Thread	XD Bit Disable (R/W)
				See Table 2-2.
		37:35		Reserved
		38	Package	Turbo Mode Disable (R/W)
				When set to 1 on processors that support Intel Turbo Boost Technology, the turbo mode feature is disabled and the IDA_Enable feature flag will be clear (CPUID.06H: EAX[1]=0).
				When set to a 0 on processors that support IDA, CPUID.06H: EAX[1] reports the processor's support of turbo mode is enabled.
				Note: The power-on default value is used by BIOS to detect hardware support of turbo mode. If the power-on default value is 1, turbo mode is available in the processor. If the power-on default value is 0, turbo mode is not available.
		63:39		Reserved
1A2H	418	MSR_TEMPERATURE_TARGET	Thread	Temperature Target
		15:0		Reserved
		23:16		Temperature Target (R)
				The minimum temperature at which PROCHOT# will be asserted. The value is degrees C.
		63:24		Reserved
1A4H	420	MSR_MISC_FEATURE_CONTROL		Miscellaneous Feature Control (R/W)
		0	Соге	L2 Hardware Prefetcher Disable (R/W)
				If 1, disables the L2 hardware prefetcher, which fetches additional lines of code or data into the L2 cache.
		1	Соге	L2 Adjacent Cache Line Prefetcher Disable (R/W)
				If 1, disables the adjacent cache line prefetcher, which fetches the cache line that comprises a cache line pair (128 bytes).
		2	Core	DCU Hardware Prefetcher Disable (R/W)
				If 1, disables the L1 data cache prefetcher, which fetches the next cache line into L1 data cache.
		3	Соге	DCU IP Prefetcher Disable (R/W)
				If 1, disables the L1 data cache IP prefetcher, which uses sequential load history (based on instruction pointer of previous loads) to determine whether to prefetch additional lines.
		63:4		Reserved
1A6H	422	MSR_OFFCORE_RSP_0	Thread	Offcore Response Event Select Register (R/W)
1AAH	426	MSR_MISC_PWR_MGMT		Miscellaneous Power Management Control
				Various model specific features enumeration. See http://biosbits.org.

Table 2-15. MSRs in Processors Based on Nehalem Microarchitecture (Contd.)

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec]		
		0	Package	EIST Hardware Coordination Disable (R/W) When O, enables hardware coordination of Enhanced Intel Speedstep Technology request from processor cores. When 1, disables hardware coordination of Enhanced Intel Speedstep Technology requests.
		1	Thread	Energy/Performance Bias Enable (R/W) This bit makes the IA32_ENERGY_PERF_BIAS register (MSR 1B0h) visible to software with Ring 0 privileges. This bit's status (1 or 0) is also reflected by CPUID.(EAX=06h):ECX[3].
		63:2		Reserved
1ACH	428	MSR_TURBO_POWER_CURRENT_LIMIT		See http://biosbits.org.
		14:0	Package	TDP Limit (R/W) TDP limit in 1/8 Watt granularity.
		15	Package	TDP Limit Override Enable (R/W) A value = 0 indicates override is not active; a value = 1 indicates override is active.
		30:16	Package	TDC Limit (R/W) TDC limit in 1/8 Amp granularity.
		31	Package	TDC Limit Override Enable (R/W) A value = 0 indicates override is not active; a value = 1 indicates override is active.
		63:32		Reserved
1ADH	429	MSR_TURBO_RATIO_LIMIT	Package	Maximum Ratio Limit of Turbo Mode R/O if MSR_PLATFORM_INFO.[28] = 0. R/W if MSR_PLATFORM_INFO.[28] = 1.
		7:0	Package	Maximum Ratio Limit for 1C Maximum turbo ratio limit of 1 core active.
		15:8	Package	Maximum Ratio Limit for 2C Maximum turbo ratio limit of 2 core active.
		23:16	Package	Maximum Ratio Limit for 3C Maximum turbo ratio limit of 3 core active.
		31:24	Package	Maximum Ratio Limit for 4C Maximum turbo ratio limit of 4 core active.
		63:32		Reserved
1C8H	456	MSR_LBR_SELECT	Core	Last Branch Record Filtering Select Register (R/W) See Section 18.9.2, "Filtering of Last Branch Records."
		0		CPL_EQ_0
		1		CPL_NEQ_0
		2		JCC
		3		NEAR_REL_CALL

Table 2-15. MSRs in Processors Based on Nehalem Microarchitecture (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		4		NEAR_IND_CALL
		5		NEAR_RET
		6		NEAR_IND_JMP
		7		NEAR_REL_JMP
		8		FAR_BRANCH
		63:9		Reserved
1C9H	457	MSR_LASTBRANCH_TOS	Thread	Last Branch Record Stack TOS (R/W)
				Contains an index (bits 0-3) that points to the MSR containing the most recent branch record.
				See MSR_LASTBRANCH_0_FROM_IP (at 680H).
1D9H	473	IA32_DEBUGCTL	Thread	Debug Control (R/W)
				See Table 2-2.
1DDH	477	MSR_LER_FROM_LIP	Thread	Last Exception Record From Linear IP (R)
				Contains a pointer to the last branch instruction that the processor executed prior to the last exception that was generated or the last interrupt that was handled.
1DEH	478	MSR_LER_TO_LIP	Thread	Last Exception Record To Linear IP (R)
				This area contains a pointer to the target of the last branch instruction that the processor executed prior to the last exception that was generated or the last interrupt that was handled.
1F2H	498	IA32_SMRR_PHYSBASE	Соге	See Table 2-2.
1F3H	499	IA32_SMRR_PHYSMASK	Core	See Table 2-2.
1FCH	508	MSR_POWER_CTL	Core	Power Control Register
				See http://biosbits.org.
		0		Reserved
		1	Package	C1E Enable (R/W)
				When set to '1', will enable the CPU to switch to the Minimum Enhanced Intel SpeedStep Technology operating point when all execution cores enter MWAIT (C1).
		63:2		Reserved
200H	512	IA32_MTRR_PHYSBASE0	Thread	See Table 2-2.
201H	513	IA32_MTRR_PHYSMASKO	Thread	See Table 2-2.
202H	514	IA32_MTRR_PHYSBASE1	Thread	See Table 2-2.
203H	515	IA32_MTRR_PHYSMASK1	Thread	See Table 2-2.
204H	516	IA32_MTRR_PHYSBASE2	Thread	See Table 2-2.
205H	517	IA32_MTRR_PHYSMASK2	Thread	See Table 2-2.
206H	518	IA32_MTRR_PHYSBASE3	Thread	See Table 2-2.

Table 2-15. MSRs in Processors Based on Nehalem Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
207H	519	IA32_MTRR_PHYSMASK3	Thread	See Table 2-2.
208H	520	IA32_MTRR_PHYSBASE4	Thread	See Table 2-2.
209H	521	IA32_MTRR_PHYSMASK4	Thread	See Table 2-2.
20AH	522	IA32_MTRR_PHYSBASE5	Thread	See Table 2-2.
20BH	523	IA32_MTRR_PHYSMASK5	Thread	See Table 2-2.
20CH	524	IA32_MTRR_PHYSBASE6	Thread	See Table 2-2.
20DH	525	IA32_MTRR_PHYSMASK6	Thread	See Table 2-2.
20EH	526	IA32_MTRR_PHYSBASE7	Thread	See Table 2-2.
20FH	527	IA32_MTRR_PHYSMASK7	Thread	See Table 2-2.
210H	528	IA32_MTRR_PHYSBASE8	Thread	See Table 2-2.
211H	529	IA32_MTRR_PHYSMASK8	Thread	See Table 2-2.
212H	530	IA32_MTRR_PHYSBASE9	Thread	See Table 2-2.
213H	531	IA32_MTRR_PHYSMASK9	Thread	See Table 2-2.
250H	592	IA32_MTRR_FIX64K_00000	Thread	See Table 2-2.
258H	600	IA32_MTRR_FIX16K_80000	Thread	See Table 2-2.
259H	601	IA32_MTRR_FIX16K_A0000	Thread	See Table 2-2.
268H	616	IA32_MTRR_FIX4K_C0000	Thread	See Table 2-2.
269H	617	IA32_MTRR_FIX4K_C8000	Thread	See Table 2-2.
26AH	618	IA32_MTRR_FIX4K_D0000	Thread	See Table 2-2.
26BH	619	IA32_MTRR_FIX4K_D8000	Thread	See Table 2-2.
26CH	620	IA32_MTRR_FIX4K_E0000	Thread	See Table 2-2.
26DH	621	IA32_MTRR_FIX4K_E8000	Thread	See Table 2-2.
26EH	622	IA32_MTRR_FIX4K_F0000	Thread	See Table 2-2.
26FH	623	IA32_MTRR_FIX4K_F8000	Thread	See Table 2-2.
277H	631	IA32_PAT	Thread	See Table 2-2.
280H	640	IA32_MC0_CTL2	Package	See Table 2-2.
281H	641	IA32_MC1_CTL2	Package	See Table 2-2.
282H	642	IA32_MC2_CTL2	Соге	See Table 2-2.
283H	643	IA32_MC3_CTL2	Соге	See Table 2-2.
284H	644	IA32_MC4_CTL2	Соге	See Table 2-2.
285H	645	IA32_MC5_CTL2	Core	See Table 2-2.
286H	646	IA32_MC6_CTL2	Package	See Table 2-2.
287H	647	IA32_MC7_CTL2	Package	See Table 2-2.
288H	648	IA32_MC8_CTL2	Package	See Table 2-2.
2FFH	767	IA32_MTRR_DEF_TYPE	Thread	Default Memory Types (R/W)
				See Table 2-2.

Table 2-15. MSRs in Processors Based on Nehalem Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
309H	777	IA32_FIXED_CTR0	Thread	Fixed-Function Performance Counter Register 0 (R/W) See Table 2-2.
30AH	778	IA32_FIXED_CTR1	Thread	Fixed-Function Performance Counter Register 1 (R/W) See Table 2-2.
30BH	779	IA32_FIXED_CTR2	Thread	Fixed-Function Performance Counter Register 2 (R/W) See Table 2-2.
345H	837	IA32_PERF_CAPABILITIES	Thread	See Table 2-2. See Section 18.4.1, "IA32_DEBUGCTL MSR."
		5:0		LBR Format See Table 2-2.
		6		PEBS Record Format
		7		PEBSSaveArchRegs
				See Table 2-2.
		11:8		PEBS_REC_FORMAT
				See Table 2-2.
		12		SMM_FREEZE
				See Table 2-2.
		63:13		Reserved
38DH	909	IA32_FIXED_CTR_CTRL	Thread	Fixed-Function-Counter Control Register (R/W)
				See Table 2-2.
38EH	910	IA32_PERF_GLOBAL_STATUS	Thread	See Table 2-2. See Section 20.6.2.2, "Global Counter Control Facilities."
38EH	910	MSR_PERF_GLOBAL_STATUS	Thread	Provides single-bit status used by software to query the overflow condition of each performance counter. (R/O)
		61		UNC_Ovf
				Uncore overflowed if 1.
38FH	911	IA32_PERF_GLOBAL_CTRL	Thread	See Table 2-2. See Section 20.6.2.2, "Global Counter Control Facilities."
390H	912	IA32_PERF_GLOBAL_OVF_CTRL	Thread	See Table 2-2. See Section 20.6.2.2, "Global Counter Control Facilities." Allows software to clear counter overflow conditions on any combination of fixed-function PMCs (IA32_FIXED_CTRx) or general-purpose PMCs via a single WRMSR.
390H	912	MSR_PERF_GLOBAL_OVF_CTRL	Thread	(R/W)
		61		CLR_UNC_Ovf
				Set 1 to clear UNC_Ovf.
3F1H	1009	IA32_PEBS_ENABLE	Thread	See Section 20.3.1.1.1, "Processor Event Based
		(MSR_PEBS_ENABLE)		Sampling (PEBS)."
		0		Enable PEBS on IA32_PMC0 (R/W)
		1		Enable PEBS on IA32_PMC1 (R/W)

Table 2-15. MSRs in Processors Based on Nehalem Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		2		Enable PEBS on IA32_PMC2 (R/W)
		3		Enable PEBS on IA32_PMC3 (R/W)
		31:4		Reserved
		32		Enable Load Latency on IA32_PMC0 (R/W)
		33		Enable Load Latency on IA32_PMC1 (R/W)
		34		Enable Load Latency on IA32_PMC2 (R/W)
		35		Enable Load Latency on IA32_PMC3 (R/W)
		63:36		Reserved
3F6H	1014	MSR_PEBS_LD_LAT	Thread	See Section 20.3.1.1.2, "Load Latency Performance Monitoring Facility."
		15:0		Minimum threshold latency value of tagged load operation that will be counted. (R/W)
		63:36		Reserved
3F8H	1016	MSR_PKG_C3_RESIDENCY	Package	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		63:0		Package C3 Residency Counter (R/O)
				Value since last reset that this package is in processor- specific C3 states. Count at the same frequency as the TSC.
3F9H	1017	MSR_PKG_C6_RESIDENCY	Package	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		63:0		Package C6 Residency Counter (R/O)
				Value since last reset that this package is in processor- specific C6 states. Count at the same frequency as the TSC.
3FAH	1018	MSR_PKG_C7_RESIDENCY	Package	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		63:0		Package C7 Residency Counter (R/O)
				Value since last reset that this package is in processor- specific C7 states. Count at the same frequency as the TSC.
3FCH	1020	MSR_CORE_C3_RESIDENCY	Core	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		63:0		CORE C3 Residency Counter (R/O)
				Value since last reset that this core is in processor- specific C3 states. Count at the same frequency as the TSC.
3FDH	1021	MSR_CORE_C6_RESIDENCY	Core	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.

Table 2-15. MSRs in Processors Based on Nehalem Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		63:0		CORE C6 Residency Counter (R/O) Value since last reset that this core is in processor- specific C6 states. Count at the same frequency as the TSC.
400H	1024	IA32_MCO_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
401H	1025	IA32_MCO_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
402H	1026	IA32_MCO_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs." The IA32_MCO_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MCO_STATUS register is clear.
				When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
403H	1027	IA32_MCO_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
404H	1028	IA32_MC1_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
405H	1029	IA32_MC1_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
406H	1030	IA32_MC1_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
				The IA32_MC1_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MC1_STATUS register is clear.
				When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
407H	1031	IA32_MC1_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
408H	1032	IA32_MC2_CTL	Соге	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
409H	1033	IA32_MC2_STATUS	Соге	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
40AH	1034	IA32_MC2_ADDR	Соге	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
				The IA32_MC2_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MC2_STATUS register is clear.
				When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
40BH	1035	IA32_MC2_MISC	Соге	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
40CH	1036	IA32_MC3_CTL	Соге	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
40DH	1037	IA32_MC3_STATUS	Соге	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
40EH	1038	IA32_MC3_ADDR	Соге	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
				The MSR_MC4_ADDR register is either not implemented or contains no address if the ADDRV flag in the MSR_MC4_STATUS register is clear.
				When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
40FH	1039	IA32_MC3_MISC	Соге	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."

Table 2-15. MSRs in Processors Based on Nehalem Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
410H	1040	IA32_MC4_CTL	Core	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
411H	1041	IA32_MC4_STATUS	Core	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
412H	1042	IA32_MC4_ADDR	Core	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
				The MSR_MC3_ADDR register is either not implemented or contains no address if the ADDRV flag in the MSR_MC3_STATUS register is clear.
				When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
413H	1043	IA32_MC4_MISC	Core	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
414H	1044	IA32_MC5_CTL	Core	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
415H	1045	IA32_MC5_STATUS	Core	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
416H	1046	IA32_MC5_ADDR	Core	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
417H	1047	IA32_MC5_MISC	Core	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
418H	1048	IA32_MC6_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
419H	1049	IA32_MC6_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
41AH	1050	IA32_MC6_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
41BH	1051	IA32_MC6_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
41CH	1052	IA32_MC7_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
41DH	1053	IA32_MC7_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
41EH	1054	IA32_MC7_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
41FH	1055	IA32_MC7_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
420H	1056	IA32_MC8_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
421H	1057	IA32_MC8_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
422H	1058	IA32_MC8_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
423H	1059	IA32_MC8_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
480H	1152	IA32_VMX_BASIC	Thread	Reporting Register of Basic VMX Capabilities (R/O)
				See Table 2-2.
				See Appendix A.1, "Basic VMX Information."
481H	1153	IA32_VMX_PINBASED_CTLS	Thread	Capability Reporting Register of Pin-based VM-execution Controls (R/O)
				See Table 2-2.
				See Appendix A.3, "VM-Execution Controls."
482H	1154	IA32_VMX_PROCBASED_CTLS	Thread	Capability Reporting Register of Primary Processor- Based VM-Execution Controls (R/O)
				See Appendix A.3, "VM-Execution Controls."

Table 2-15. MSRs in Processors Based on Nehalem Microarchitecture (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
483H	1155	IA32_VMX_EXIT_CTLS	Thread	Capability Reporting Register of VM-Exit Controls (R/O) See Table 2-2.
				See Appendix A.4, "VM-Exit Controls."
484H	1156	IA32_VMX_ENTRY_CTLS	Thread	Capability Reporting Register of VM-Entry Controls (R/O)
				See Table 2-2.
				See Appendix A.5, "VM-Entry Controls."
485H	1157	IA32_VMX_MISC	Thread	Reporting Register of Miscellaneous VMX Capabilities (R/O)
				See Table 2-2.
				See Appendix A.6, "Miscellaneous Data."
486H	1158	IA32_VMX_CRO_FIXEDO	Thread	Capability Reporting Register of CRO Bits Fixed to 0 (R/O)
				See Table 2-2.
				See Appendix A.7, "VMX-Fixed Bits in CR0."
487H	1159	IA32_VMX_CRO_FIXED1	Thread	Capability Reporting Register of CRO Bits Fixed to 1 (R/O)
				See Table 2-2.
				See Appendix A.7, "VMX-Fixed Bits in CR0."
488H	1160	IA32_VMX_CR4_FIXED0	Thread	Capability Reporting Register of CR4 Bits Fixed to 0 (R/O)
				See Table 2-2.
				See Appendix A.8, "VMX-Fixed Bits in CR4."
489H	1161	IA32_VMX_CR4_FIXED1	Thread	Capability Reporting Register of CR4 Bits Fixed to 1 (R/O)
				See Table 2-2.
				See Appendix A.8, "VMX-Fixed Bits in CR4."
48AH	1162	IA32_VMX_VMCS_ENUM	Thread	Capability Reporting Register of VMCS Field Enumeration (R/O)
				See Table 2-2.
				See Appendix A.9, "VMCS Enumeration."
48BH	1163	IA32_VMX_PROCBASED_CTLS2	Thread	Capability Reporting Register of Secondary Processor- Based VM-Execution Controls (R/O)
				See Appendix A.3, "VM-Execution Controls."
600H	1536	IA32_DS_AREA	Thread	DS Save Area (R/W)
				See Table 2-2.
				See Section 20.6.3.4, "Debug Store (DS) Mechanism."

Table 2-15. MSRs in Processors Based on Nehalem Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
680H	1664	MSR_LASTBRANCH_0_FROM_IP	Thread	Last Branch Record O From IP (R/W)
				One of sixteen pairs of last branch record registers on the last branch record stack. The From_IP part of the stack contains pointers to the source instruction. See also:
				 Last Branch Record Stack TOS at 1C9H. Section 18.9.1 and record format in Section 18.4.8.1.
681H	1665	MSR_LASTBRANCH_1_FROM_IP	Thread	Last Branch Record 1 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
682H	1666	MSR_LASTBRANCH_2_FROM_IP	Thread	Last Branch Record 2 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
683H	1667	MSR_LASTBRANCH_3_FROM_IP	Thread	Last Branch Record 3 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
684H	1668	MSR_LASTBRANCH_4_FROM_IP	Thread	Last Branch Record 4 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
685H	1669	MSR_LASTBRANCH_5_FROM_IP	Thread	Last Branch Record 5 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
686H	1670	MSR_LASTBRANCH_6_FROM_IP	Thread	Last Branch Record 6 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
687H	1671	MSR_LASTBRANCH_7_FROM_IP	Thread	Last Branch Record 7 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
688H	1672	MSR_LASTBRANCH_8_FROM_IP	Thread	Last Branch Record 8 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
689H	1673	MSR_LASTBRANCH_9_FROM_IP	Thread	Last Branch Record 9 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
68AH	1674	MSR_LASTBRANCH_10_FROM_IP	Thread	Last Branch Record 10 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
68BH	1675	MSR_LASTBRANCH_11_FROM_IP	Thread	Last Branch Record 11 From IP (R/W)
	1.555			See description of MSR_LASTBRANCH_0_FROM_IP.
68CH	1676	MSR_LASTBRANCH_12_FROM_IP	Thread	Last Branch Record 12 From IP (R/W)
CODII	1677	MCD LACTEDANICH 12 CDOM ID		See description of MSR_LASTBRANCH_0_FROM_IP.
68DH	1677	MSR_LASTBRANCH_13_FROM_IP	Thread	Last Branch Record 13 From IP (R/W)
COCII	1670	MCD LACTEDANICH 14 CDOM ID	TI	See description of MSR_LASTBRANCH_0_FROM_IP.
68EH	1678	MSR_LASTBRANCH_14_FROM_IP	Thread	Last Branch Record 14 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
68FH	1670	MSD LASTRDANICH 15 CDOM ID	Throad	·
UOFF	1679	MSR_LASTBRANCH_15_FROM_IP	Thread	Last Branch Record 15 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
6C0H	1728	MSR_LASTBRANCH_0_T0_IP	Thread	Last Branch Record 0 To IP (R/W)
осоп	1720	I ISIN_CAS I DIVANICIT_U_TU_IF	THEOU	One of sixteen pairs of last branch record registers on the last branch record stack. This part of the stack contains pointers to the destination instruction.

Table 2-15. MSRs in Processors Based on Nehalem Microarchitecture (Contd.)

Hex Dec		ister ress	Register Name / Bit Fields	Scope	Bit Description
See description of MSR_LASTBRANCH_0_TO_IP	Hex	Dec			
1730 MSR_LASTBRANCH_2_TO_IP	6C1H	1729	MSR_LASTBRANCH_1_TO_IP	Thread	Last Branch Record 1 To IP (R/W)
See description of MSR_LASTBRANCH_0_TO_JP. 6C3H 1731 MSR_LASTBRANCH_3_TO_JP Thread Last Branch Record 3 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_JP. 6C4H 1732 MSR_LASTBRANCH_4_TO_JP Thread Last Branch Record 4 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_JP. 6C5H 1733 MSR_LASTBRANCH_5_TO_JP Thread Last Branch Record 5 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_JP. 6C6H 1734 MSR_LASTBRANCH_6_TO_JP Thread Last Branch Record 6 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_JP. 6C7H 1735 MSR_LASTBRANCH_7_TO_JP Thread Last Branch Record 6 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_JP. 6C8H 1736 MSR_LASTBRANCH_8_TO_JP Thread Last Branch Record 7 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_JP. 6C9H 1737 MSR_LASTBRANCH_9_TO_JP Thread Last Branch Record 8 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_JP. 6C6H 1738 MSR_LASTBRANCH_10_TO_JP Thread Last Branch Record 9 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_JP. 6C6H 1739 MSR_LASTBRANCH_10_TO_JP Thread Last Branch Record 10 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_JP. 6C6H 1740 MSR_LASTBRANCH_11_TO_JP Thread Last Branch Record 11 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_JP. 6C6H 1741 MSR_LASTBRANCH_12_TO_JP Thread Last Branch Record 11 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_JP. 6C6H 1742 MSR_LASTBRANCH_13_TO_JP Thread Last Branch Record 12 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_JP. 6C6H 1742 MSR_LASTBRANCH_13_TO_JP Thread Last Branch Record 13 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_JP. 6C6H 1742 MSR_LASTBRANCH_15_TO_JP Thread Last Branch Record 13 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_JP. 6C6H 1742 MSR_LASTBRANCH_15_TO_JP Thread Last Branch Record 13 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_JP. 6C6H 1742 MSR_LASTBRANCH_15_TO_JP Thread Last Branch Record 15 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_JP. 6C7H 1743 MSR_LASTBRANCH_15_TO_JP Thread Last Branch Record 15 To IP (R/W) See description of MSR_LASTBRANCH_0_TO					See description of MSR_LASTBRANCH_0_T0_IP.
1731 MSR_LASTBRANCH_3_TO_IP Thread Last Branch Record 3 To IP (R/W)	6C2H	1730	MSR_LASTBRANCH_2_TO_IP	Thread	Last Branch Record 2 To IP (R/W)
See description of MSR_LASTBRANCH_0_TO_JP. 6C4H 1732 MSR_LASTBRANCH_4_TO_JP Thread Last Branch Record 4 To IP (R/W) 5ee description of MSR_LASTBRANCH_0_TO_JP. 6C5H 1733 MSR_LASTBRANCH_5_TO_JP Thread Last Branch Record 5 To IP (R/W) 6C6H 1734 MSR_LASTBRANCH_6_TO_JP Thread Last Branch Record 5 To IP (R/W) 6C6H 1735 MSR_LASTBRANCH_7_TO_JP Thread Last Branch Record 6 To IP (R/W) 6C7H 1735 MSR_LASTBRANCH_7_TO_JP Thread Last Branch Record 7 To IP (R/W) 6C8H 1736 MSR_LASTBRANCH_8_TO_JP Thread Last Branch Record 8 To IP (R/W) 6C9H 1737 MSR_LASTBRANCH_9_TO_JP Thread Last Branch Record 8 To IP (R/W) 6C9H 1738 MSR_LASTBRANCH_9_TO_JP Thread Last Branch Record 8 To IP (R/W) 6C9H 1738 MSR_LASTBRANCH_10_TO_JP Thread Last Branch Record 10 To IP (R/W) 6C9H 1739 MSR_LASTBRANCH_10_TO_JP Thread Last Branch Record 10 To IP (R/W) 6C9H 1739 MSR_LASTBRANCH_11_TO_JP Thread Last Branch Record 10 To IP (R/W) 6C9H 1739 MSR_LASTBRANCH_11_TO_JP Thread Last Branch Record 10 To IP (R/W) 6C9H 1739 MSR_LASTBRANCH_11_TO_JP Thread Last Branch Record 11 To IP (R/W) 6C9H 1740 MSR_LASTBRANCH_11_TO_JP Thread Last Branch Record 11 To IP (R/W) 6C9H 1741 MSR_LASTBRANCH_12_TO_JP Thread Last Branch Record 12 To IP (R/W) 6C9H 1742 MSR_LASTBRANCH_13_TO_JP Thread Last Branch Record 13 To IP (R/W) 6C9H 1742 MSR_LASTBRANCH_14_TO_JP Thread Last Branch Record 15 To IP (R/W) 6C9H 1743 MSR_LASTBRANCH_14_TO_JP Thread Last Branch Record 15 To IP (R/W) 6C9H 1744 MSR_LASTBRANCH_15_TO_JP Thread Last Branch Record 16 MSR_LASTBRANCH_0_TO_JP. 6C9H 1749 MSR_LASTBRANCH_15_TO_JP Thread Last Branch Record 17 To IP (R/W) 6C9H 1740 MSR_LASTBRANCH_15_TO_JP Thread Last Branch Record 17 To IP (R/W) 6C9H 1740 MSR_LASTBRANCH_15_TO_JP Thread Last Branch Record 17 To IP (R/W) 6C9H 1741 MSR_LASTBRANCH_15_TO_JP Thread Last Branch Record 18 To IP (R/W) 6C9H 1742 MSR_LASTBRANCH_15_TO_JP Thread X2APIC DRegister (R/O) 6C9H 1742 MSR_LASTBRANCH_15_TO_JP Thread X2APIC DREgister (R/O) 6C9H 1744 MSR_LASTBRANCH_15_TO_JP Thread X2APIC DREgister (R/O) 6C9H 1744 MSR_LASTB					See description of MSR_LASTBRANCH_0_T0_IP.
C4H	6C3H	1731	MSR_LASTBRANCH_3_TO_IP	Thread	Last Branch Record 3 To IP (R/W)
See description of MSR_LASTBRANCH_0_T0_IP. 6C5H 1733 MSR_LASTBRANCH_5_T0_IP Thread Last Branch Record 5 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 6C6H 1734 MSR_LASTBRANCH_6_T0_IP Thread Last Branch Record 6 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 6C7H 1735 MSR_LASTBRANCH_7_T0_IP Thread Last Branch Record 7 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 6C8H 1736 MSR_LASTBRANCH_8_T0_IP Thread Last Branch Record 8 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 6C9H 1737 MSR_LASTBRANCH_9_T0_IP Thread Last Branch Record 9 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 6C8H 1738 MSR_LASTBRANCH_10_T0_IP Thread Last Branch Record 10 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 6C8H 1739 MSR_LASTBRANCH_11_T0_IP Thread Last Branch Record 11 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 6C8H 1740 MSR_LASTBRANCH_11_T0_IP Thread Last Branch Record 12 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 6C9H 1741 MSR_LASTBRANCH_12_T0_IP Thread Last Branch Record 12 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 6C9H 1741 MSR_LASTBRANCH_13_T0_IP Thread Last Branch Record 12 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 6C9H 1742 MSR_LASTBRANCH_13_T0_IP Thread Last Branch Record 12 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 6C9H 1743 MSR_LASTBRANCH_14_T0_IP Thread Last Branch Record 14 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 6C9H 1743 MSR_LASTBRANCH_15_T0_IP Thread Last Branch Record 15 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 6C9H 1743 MSR_LASTBRANCH_15_T0_IP Thread Last Branch Record 15 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 6C9H 1743 MSR_LASTBRANCH_15_T0_IP Thread Last Branch Record 15 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 6C9H 1743 MSR_LASTBRANCH_15_T0_IP Thread Last Branch Record 15 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 6C9H 1743 MSR_LASTBRANCH_0_T0_IP. 6C9H 1744 MSR_LASTBRANCH_0_T0_IP. 6C9H 1745 MSR_LASTBRANCH_0_T0_IP. 6C9					See description of MSR_LASTBRANCH_0_T0_IP.
SC5H	6C4H	1732	MSR_LASTBRANCH_4_TO_IP	Thread	, ,
See description of MSR_LASTBRANCH_0_TO_IP.					See description of MSR_LASTBRANCH_0_T0_IP.
CGCH	6C5H	1733	MSR_LASTBRANCH_5_TO_IP	Thread	, ,
See description of MSR_LASTBRANCH_0_TO_IP. 6C7H 1735 MSR_LASTBRANCH_0_TO_IP Thread Last Branch Record 7 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6C8H 1736 MSR_LASTBRANCH_8_TO_IP Thread Last Branch Record 8 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6C9H 1737 MSR_LASTBRANCH_9_TO_IP Thread Last Branch Record 9 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CAH 1738 MSR_LASTBRANCH_10_TO_IP Thread Last Branch Record 10 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CBH 1739 MSR_LASTBRANCH_11_TO_IP Thread Last Branch Record 11 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CCH 1740 MSR_LASTBRANCH_12_TO_IP Thread Last Branch Record 12 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CCH 1741 MSR_LASTBRANCH_13_TO_IP Thread Last Branch Record 13 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CCH 1742 MSR_LASTBRANCH_13_TO_IP Thread Last Branch Record 13 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CCH 1742 MSR_LASTBRANCH_14_TO_IP Thread Last Branch Record 14 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CFH 1743 MSR_LASTBRANCH_14_TO_IP Thread Last Branch Record 14 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CFH 1743 MSR_LASTBRANCH_15_TO_IP Thread Last Branch Record 15 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 802H 2050 IA32_X2APIC_APICID Thread X2APIC ID Register (R/O) 803H 2051 IA32_X2APIC_VERSION Thread X2APIC ID Register (R/O) 808H 2058 IA32_X2APIC_VERSION Thread X2APIC Task Priority Register (R/W) 80AH 2058 IA32_X2APIC_EPR Thread X2APIC EDI Register (R/O) 80BH 2059 IA32_X2APIC_EDI Thread X2APIC Degical Destination Register (R/W) 80CH 2063 IA32_X2APIC_EDI Thread X2APIC Spurious Interrupt Vector Register (R/W) 81DH 2064 IA32_X2APIC_ISRO Thread X2APIC Spurious Interrupt Vector Register (R/W)					See description of MSR_LASTBRANCH_0_T0_IP.
Thread	6C6H	1734	MSR_LASTBRANCH_6_TO_IP	Thread	, ,
See description of MSR_LASTBRANCH_0_TO_IP.					See description of MSR_LASTBRANCH_0_T0_IP.
Thread	6C7H	1735	MSR_LASTBRANCH_7_TO_IP	Thread	, ,
See description of MSR_LASTBRANCH_0_T0_IP.					
Thread	6C8H	1736	MSR_LASTBRANCH_8_TO_IP	Thread	, ,
See description of MSR_LASTBRANCH_0_TO_IP. 6CAH 1738 MSR_LASTBRANCH_10_TO_IP Thread Last Branch Record 10 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CBH 1739 MSR_LASTBRANCH_11_TO_IP Thread Last Branch Record 11 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CCH 1740 MSR_LASTBRANCH_12_TO_IP Thread Last Branch Record 12 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CDH 1741 MSR_LASTBRANCH_13_TO_IP Thread Last Branch Record 13 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CEH 1742 MSR_LASTBRANCH_14_TO_IP Thread Last Branch Record 14 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CFH 1743 MSR_LASTBRANCH_15_TO_IP Thread Last Branch Record 15 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 802H 2050 IA32_X2APIC_APICID Thread X2APIC ID Register (R/O) 803H 2051 IA32_X2APIC_VERSION Thread X2APIC Usersion Register (R/W) 80AH 2058 IA32_X2APIC_PPR Thread X2APIC Task Priority Register (R/W) 80AH 2059 IA32_X2APIC_PPR Thread X2APIC DIREGISTER (R/O) 80BH 2050 IA32_X2APIC_E0I Thread X2APIC E0I Register (W/O) 80BH 2051 IA32_X2APIC_EDI Thread X2APIC E0I Register (W/O) 80BH 2051 IA32_X2APIC_EDI Thread X2APIC E0I Register (W/O) 80BH 2051 IA32_X2APIC_EDI Thread X2APIC E0I Register (R/W) 80BH 2051 IA32_X2APIC_EDI Thread X2APIC E0I Register (R/O)					·
6CAH 1738 MSR_LASTBRANCH_10_TO_IP Thread Last Branch Record 10 To IP (R/W) 6CBH 1739 MSR_LASTBRANCH_11_TO_IP Thread Last Branch Record 11 To IP (R/W) 6CBH 1739 MSR_LASTBRANCH_11_TO_IP Thread Last Branch Record 11 To IP (R/W) 6CCH 1740 MSR_LASTBRANCH_12_TO_IP Thread Last Branch Record 12 To IP (R/W) 6CDH 1741 MSR_LASTBRANCH_13_TO_IP Thread Last Branch Record 13 To IP (R/W) 6CEH 1742 MSR_LASTBRANCH_14_TO_IP Thread Last Branch Record 14 To IP (R/W) 6CFH 1743 MSR_LASTBRANCH_15_TO_IP Thread Last Branch Record 15 To IP (R/W) 6CFH 1743 MSR_LASTBRANCH_15_TO_IP Thread Last Branch Record 15 To IP (R/W) 6CFH 1743 MSR_LASTBRANCH_15_TO_IP Thread Last Branch Record 15 To IP (R/W) 802H 2050 IA32_X2APIC_APICID Thread X2APIC ID Register (R/O) 803H 2051 IA32_X2APIC_EPIC Thread X2APIC Task Priority Register (R/W) 804H 2058 IA32_X2APIC_EDI	6C9H	1737	MSR_LASTBRANCH_9_TO_IP	Thread	, , ,
See description of MSR_LASTBRANCH_0_TO_IP. 6CBH 1739 MSR_LASTBRANCH_11_TO_IP Thread Last Branch Record 11 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CCH 1740 MSR_LASTBRANCH_12_TO_IP Thread Last Branch Record 12 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CDH 1741 MSR_LASTBRANCH_13_TO_IP Thread Last Branch Record 13 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CEH 1742 MSR_LASTBRANCH_14_TO_IP Thread Last Branch Record 14 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CFH 1743 MSR_LASTBRANCH_15_TO_IP Thread Last Branch Record 15 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 802H 2050 IA32_X2APIC_APICID Thread x2APIC ID Register (R/O) 803H 2051 IA32_X2APIC_VERSION Thread x2APIC DRegister (R/O) 808H 2056 IA32_X2APIC_TPR Thread x2APIC Task Priority Register (R/W) 80AH 2058 IA32_X2APIC_EOI Thread x2APIC FOrcessor Priority Register (R/O) 80BH 2050 IA32_X2APIC_EOI Thread x2APIC EOI Register (W/O) 80BH 2051 IA32_X2APIC_EOI Thread x2APIC EOI Register (W/O) 80BH 2053 IA32_X2APIC_EOI Thread x2APIC EOI Register (W/O) 80BH 2054 IA32_X2APIC_EOI Thread x2APIC EOI Register (W/O) 80BH 2056 IA32_X2APIC_EOI Thread x2APIC EOI Register (W/O) 80BH 2056 IA32_X2APIC_EOI Thread x2APIC EOI Register (R/O) 80BH 2056 IA32_X2APIC_EOI Thread x2APIC Spurious Interrupt Vector Register (R/W) 810H 2064 IA32_X2APIC_ISRO Thread x2APIC In-Service Register Bits [31:0] (R/O)					·
6CBH 1739 MSR_LASTBRANCH_11_TO_IP Thread Last Branch Record 11 To IP (R/W) 6CCH 1740 MSR_LASTBRANCH_12_TO_IP Thread Last Branch Record 12 To IP (R/W) 6CCH 1740 MSR_LASTBRANCH_13_TO_IP Thread Last Branch Record 12 To IP (R/W) 6CDH 1741 MSR_LASTBRANCH_13_TO_IP Thread Last Branch Record 13 To IP (R/W) 6CEH 1742 MSR_LASTBRANCH_14_TO_IP Thread Last Branch Record 14 To IP (R/W) 6CFH 1743 MSR_LASTBRANCH_15_TO_IP Thread Last Branch Record 15 To IP (R/W) 6CFH 1743 MSR_LASTBRANCH_15_TO_IP Thread Last Branch Record 15 To IP (R/W) 802H 2050 IA32_X2APIC_APICID Thread X2APIC ID Register (R/O) 803H 2051 IA32_X2APIC_VERSION Thread X2APIC Version Register (R/O) 808H 2050 IA32_X2APIC_TPR Thread X2APIC Task Priority Register (R/O) 808H 2059 IA32_X2APIC_PPR Thread X2APIC EOI Register (W/O) 80BH 2059 IA32_X2APIC_EOI Thread <	6CAH	1738	MSR_LASTBRANCH_10_TO_IP	Thread	, ,
See description of MSR_LASTBRANCH_0_TO_IP. 6CCH 1740 MSR_LASTBRANCH_12_TO_IP Thread Last Branch Record 12 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CDH 1741 MSR_LASTBRANCH_13_TO_IP Thread Last Branch Record 13 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CCH 1742 MSR_LASTBRANCH_14_TO_IP Thread Last Branch Record 14 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CCH 1743 MSR_LASTBRANCH_15_TO_IP Thread Last Branch Record 15 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CCH 1743 MSR_LASTBRANCH_15_TO_IP Thread Last Branch Record 15 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 802H 2050 IA32_X2APIC_APICID Thread x2APIC ID Register (R/O) 803H 2051 IA32_X2APIC_VERSION Thread x2APIC Version Register (R/W) 808H 2056 IA32_X2APIC_TPR Thread x2APIC Task Priority Register (R/W) 80AH 2058 IA32_X2APIC_PPR Thread x2APIC Processor Priority Register (R/O) 80BH 2059 IA32_X2APIC_EOI Thread x2APIC EOI Register (W/O) 80BH 2061 IA32_X2APIC_EOI Thread x2APIC EOI Register (W/O) 80BH 2063 IA32_X2APIC_LDR Thread x2APIC Spurious Interrupt Vector Register (R/W) 810H 2064 IA32_X2APIC_ISRO Thread x2APIC In-Service Register Bits [31:0] (R/O)	CCDLI	1720	MCD LACTEDANICH 11 TO ID		·
6CCH1740MSR_LASTBRANCH_12_TO_IPThreadLast Branch Record 12 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.6CDH1741MSR_LASTBRANCH_13_TO_IPThreadLast Branch Record 13 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.6CEH1742MSR_LASTBRANCH_14_TO_IPThreadLast Branch Record 14 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.6CFH1743MSR_LASTBRANCH_15_TO_IPThreadLast Branch Record 15 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.802H2050IA32_X2APIC_APICIDThreadx2APIC ID Register (R/O)803H2051IA32_X2APIC_VERSIONThreadx2APIC Version Register (R/O)808H2056IA32_X2APIC_TPRThreadx2APIC Task Priority Register (R/W)80AH2058IA32_X2APIC_PPRThreadx2APIC Processor Priority Register (R/O)80BH2059IA32_X2APIC_EOIThreadx2APIC EOI Register (W/O)80DH2061IA32_X2APIC_LDRThreadx2APIC Logical Destination Register (R/O)80FH2063IA32_X2APIC_SIVRThreadx2APIC Spurious Interrupt Vector Register (R/W)810H2064IA32_X2APIC_ISROThreadx2APIC In-Service Register Bits [31:0] (R/O)	9CBH	1/39	MSR_LASTBRANCH_TT_TO_IP	Thread	, ,
See description of MSR_LASTBRANCH_0_T0_IP. 6CDH 1741 MSR_LASTBRANCH_13_T0_IP Thread Last Branch Record 13 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 6CEH 1742 MSR_LASTBRANCH_14_T0_IP Thread Last Branch Record 14 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 6CFH 1743 MSR_LASTBRANCH_15_T0_IP Thread Last Branch Record 15 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 802H 2050 IA32_X2APIC_APICID Thread X2APIC ID Register (R/O) 803H 2051 IA32_X2APIC_VERSION Thread X2APIC Version Register (R/O) 808H 2056 IA32_X2APIC_TPR Thread X2APIC Task Priority Register (R/W) 80AH 2058 IA32_X2APIC_PPR Thread X2APIC Processor Priority Register (R/O) 80BH 2059 IA32_X2APIC_EOI Thread X2APIC EOI Register (W/O) 80BH 2061 IA32_X2APIC_LDR Thread X2APIC Logical Destination Register (R/O) 80FH 2063 IA32_X2APIC_SIVR Thread X2APIC Spurious Interrupt Vector Register (R/W) 810H 2064 IA32_X2APIC_ISRO Thread X2APIC In-Service Register Bits [31:0] (R/O)	CCCLL	1740	MCD LACTEDANICH 12 TO ID	T1 1	·
Thread	БССН	1740	MSK_CASTBRANCH_T2_TO_IP	Inread	` ,
See description of MSR_LASTBRANCH_0_TO_IP. 6CEH 1742 MSR_LASTBRANCH_14_TO_IP Thread Last Branch Record 14 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CFH 1743 MSR_LASTBRANCH_15_TO_IP Thread Last Branch Record 15 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 802H 2050 IA32_X2APIC_APICID Thread X2APIC ID Register (R/O) 803H 2051 IA32_X2APIC_VERSION Thread X2APIC Version Register (R/O) 808H 2056 IA32_X2APIC_TPR Thread X2APIC Task Priority Register (R/W) 80AH 2058 IA32_X2APIC_PPR Thread X2APIC Frocessor Priority Register (R/O) 80BH 2059 IA32_X2APIC_EOI Thread X2APIC EOI Register (W/O) 80DH 2061 IA32_X2APIC_LDR Thread X2APIC Logical Destination Register (R/O) 80FH 2063 IA32_X2APIC_SIVR Thread X2APIC Spurious Interrupt Vector Register (R/W) 810H 2064 IA32_X2APIC_ISRO Thread X2APIC In-Service Register Bits [31:0] (R/O)	CCDII	1741	MSD LASTDDANICH 12 TO ID	Throad	·
6CEH 1742 MSR_LASTBRANCH_14_TO_IP Thread Last Branch Record 14 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 6CFH 1743 MSR_LASTBRANCH_15_TO_IP Thread Last Branch Record 15 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 802H 2050 IA32_X2APIC_APICID Thread x2APIC ID Register (R/O) 803H 2051 IA32_X2APIC_VERSION Thread x2APIC Version Register (R/O) 808H 2056 IA32_X2APIC_TPR Thread x2APIC Task Priority Register (R/W) 80AH 2058 IA32_X2APIC_PPR Thread x2APIC Processor Priority Register (R/O) 80BH 2059 IA32_X2APIC_EOI Thread x2APIC EOI Register (W/O) 80DH 2061 IA32_X2APIC_LDR Thread x2APIC Logical Destination Register (R/O) 80FH 2063 IA32_X2APIC_SIVR Thread x2APIC Spurious Interrupt Vector Register (R/W) 810H 2064 IA32_X2APIC_ISRO Thread x2APIC In-Service Register Bits [31:0] (R/O)	осып	1741	MSK_CASTBRANCH_TS_TO_IP	IIIIeau	` ,
See description of MSR_LASTBRANCH_0_T0_IP. 6CFH 1743 MSR_LASTBRANCH_15_T0_IP Thread Last Branch Record 15 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP. 802H 2050 IA32_X2APIC_APICID Thread x2APIC ID Register (R/O) 803H 2051 IA32_X2APIC_VERSION Thread x2APIC Version Register (R/O) 808H 2056 IA32_X2APIC_TPR Thread x2APIC Task Priority Register (R/W) 80AH 2058 IA32_X2APIC_PPR Thread x2APIC Processor Priority Register (R/O) 80BH 2059 IA32_X2APIC_EOI Thread x2APIC EOI Register (W/O) 80DH 2061 IA32_X2APIC_LDR Thread x2APIC Logical Destination Register (R/O) 80FH 2063 IA32_X2APIC_SIVR Thread x2APIC Spurious Interrupt Vector Register (R/W) 810H 2064 IA32_X2APIC_ISRO Thread x2APIC In-Service Register Bits [31:0] (R/O)	ECEH .	17/12	MSD LASTRDANICH 14 TO ID	Thread	·
6CFH 1743 MSR_LASTBRANCH_15_TO_IP Thread Last Branch Record 15 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP. 802H 2050 IA32_X2APIC_APICID Thread X2APIC ID Register (R/O) 803H 2051 IA32_X2APIC_VERSION Thread X2APIC Version Register (R/O) 808H 2056 IA32_X2APIC_TPR Thread X2APIC Task Priority Register (R/W) 80AH 2058 IA32_X2APIC_PPR Thread X2APIC Processor Priority Register (R/O) 80BH 2059 IA32_X2APIC_EOI Thread X2APIC EOI Register (W/O) 80DH 2061 IA32_X2APIC_LDR Thread X2APIC Logical Destination Register (R/O) 80FH 2063 IA32_X2APIC_SIVR Thread X2APIC Spurious Interrupt Vector Register (R/W) 810H 2064 IA32_X2APIC_ISRO Thread X2APIC In-Service Register Bits [31:0] (R/O)	OCCIT	1742	TISIC_CAST BIXANCII_14_10_II	IIIIeau	, ,
See description of MSR_LASTBRANCH_0_TO_IP. 802H 2050 IA32_X2APIC_APICID Thread X2APIC ID Register (R/O) 803H 2051 IA32_X2APIC_VERSION Thread X2APIC Version Register (R/O) 808H 2056 IA32_X2APIC_TPR Thread X2APIC Task Priority Register (R/W) 80AH 2058 IA32_X2APIC_PPR Thread X2APIC Processor Priority Register (R/O) 80BH 2059 IA32_X2APIC_EOI Thread X2APIC EOI Register (W/O) 80DH 2061 IA32_X2APIC_LDR Thread X2APIC Logical Destination Register (R/O) 80FH 2063 IA32_X2APIC_SIVR Thread X2APIC Spurious Interrupt Vector Register (R/W) 810H 2064 IA32_X2APIC_ISRO Thread X2APIC In-Service Register Bits [31:0] (R/O)	6CEH	1743	MSR LASTRRANCH 15 TO IP	Thread	·
802H2050IA32_X2APIC_APICIDThreadx2APIC ID Register (R/O)803H2051IA32_X2APIC_VERSIONThreadx2APIC Version Register (R/O)808H2056IA32_X2APIC_TPRThreadx2APIC Task Priority Register (R/W)80AH2058IA32_X2APIC_PPRThreadx2APIC Processor Priority Register (R/O)80BH2059IA32_X2APIC_EOIThreadx2APIC EOI Register (W/O)80DH2061IA32_X2APIC_LDRThreadx2APIC Logical Destination Register (R/O)80FH2063IA32_X2APIC_SIVRThreadx2APIC Spurious Interrupt Vector Register (R/W)810H2064IA32_X2APIC_ISROThreadx2APIC In-Service Register Bits [31:0] (R/O)	Jein	17 13	TISK_CASTBINITER_TS_TS_II	1111000	, ,
803H2051IA32_X2APIC_VERSIONThreadx2APIC Version Register (R/O)808H2056IA32_X2APIC_TPRThreadx2APIC Task Priority Register (R/W)80AH2058IA32_X2APIC_PPRThreadx2APIC Processor Priority Register (R/O)80BH2059IA32_X2APIC_EOIThreadx2APIC EOI Register (W/O)80DH2061IA32_X2APIC_LDRThreadx2APIC Logical Destination Register (R/O)80FH2063IA32_X2APIC_SIVRThreadx2APIC Spurious Interrupt Vector Register (R/W)810H2064IA32_X2APIC_ISROThreadx2APIC In-Service Register Bits [31:0] (R/O)	802H	2050	IA32 X2APIC APICID	Thread	
808H2056IA32_X2APIC_TPRThreadx2APIC Task Priority Register (R/W)80AH2058IA32_X2APIC_PPRThreadx2APIC Processor Priority Register (R/O)80BH2059IA32_X2APIC_E0IThreadx2APIC E0I Register (W/O)80DH2061IA32_X2APIC_LDRThreadx2APIC Logical Destination Register (R/O)80FH2063IA32_X2APIC_SIVRThreadx2APIC Spurious Interrupt Vector Register (R/W)810H2064IA32_X2APIC_ISROThreadx2APIC In-Service Register Bits [31:0] (R/O)					
80AH2058IA32_X2APIC_PPRThreadx2APIC Processor Priority Register (R/O)80BH2059IA32_X2APIC_EOIThreadx2APIC EOI Register (W/O)80DH2061IA32_X2APIC_LDRThreadx2APIC Logical Destination Register (R/O)80FH2063IA32_X2APIC_SIVRThreadx2APIC Spurious Interrupt Vector Register (R/W)810H2064IA32_X2APIC_ISROThreadx2APIC In-Service Register Bits [31:0] (R/O)					- , ,
80BH2059IA32_X2APIC_EOIThreadx2APIC EOI Register (W/O)80DH2061IA32_X2APIC_LDRThreadx2APIC Logical Destination Register (R/O)80FH2063IA32_X2APIC_SIVRThreadx2APIC Spurious Interrupt Vector Register (R/W)810H2064IA32_X2APIC_ISROThreadx2APIC In-Service Register Bits [31:0] (R/O)					
80DH 2061 IA32_X2APIC_LDR Thread x2APIC Logical Destination Register (R/O) 80FH 2063 IA32_X2APIC_SIVR Thread x2APIC Spurious Interrupt Vector Register (R/W) 810H 2064 IA32_X2APIC_ISRO Thread x2APIC In-Service Register Bits [31:0] (R/O)					3 3 ()
80FH 2063 IA32_X2APIC_SIVR Thread x2APIC Spurious Interrupt Vector Register (R/W) 810H 2064 IA32_X2APIC_ISR0 Thread x2APIC In-Service Register Bits [31:0] (R/O)					<u> </u>
810H 2064 IA32_X2APIC_ISR0 Thread x2APIC In-Service Register Bits [31:0] (R/0)					
811H 2065 IA32_X2APIC_ISR1 Thread x2APIC In-Service Register Bits [63:32] (R/O)					

Table 2-15. MSRs in Processors Based on Nehalem Microarchitecture (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
812H	2066	IA32_X2APIC_ISR2	Thread	x2APIC In-Service Register Bits [95:64] (R/0)
813H	2067	IA32_X2APIC_ISR3	Thread	x2APIC In-Service Register Bits [127:96] (R/0)
814H	2068	IA32_X2APIC_ISR4	Thread	x2APIC In-Service Register Bits [159:128] (R/0)
815H	2069	IA32_X2APIC_ISR5	Thread	x2APIC In-Service Register Bits [191:160] (R/0)
816H	2070	IA32_X2APIC_ISR6	Thread	x2APIC In-Service Register Bits [223:192] (R/0)
817H	2071	IA32_X2APIC_ISR7	Thread	x2APIC In-Service Register Bits [255:224] (R/0)
818H	2072	IA32_X2APIC_TMR0	Thread	x2APIC Trigger Mode Register Bits [31:0] (R/O)
819H	2073	IA32_X2APIC_TMR1	Thread	x2APIC Trigger Mode Register Bits [63:32] (R/O)
81AH	2074	IA32_X2APIC_TMR2	Thread	x2APIC Trigger Mode Register Bits [95:64] (R/O)
81BH	2075	IA32_X2APIC_TMR3	Thread	x2APIC Trigger Mode Register Bits [127:96] (R/O)
81CH	2076	IA32_X2APIC_TMR4	Thread	x2APIC Trigger Mode Register Bits [159:128] (R/O)
81DH	2077	IA32_X2APIC_TMR5	Thread	x2APIC Trigger Mode Register Bits [191:160] (R/O)
81EH	2078	IA32_X2APIC_TMR6	Thread	x2APIC Trigger Mode Register Bits [223:192] (R/O)
81FH	2079	IA32_X2APIC_TMR7	Thread	x2APIC Trigger Mode Register Bits [255:224] (R/O)
820H	2080	IA32_X2APIC_IRR0	Thread	x2APIC Interrupt Request Register Bits [31:0] (R/0)
821H	2081	IA32_X2APIC_IRR1	Thread	x2APIC Interrupt Request Register Bits [63:32] (R/0)
822H	2082	IA32_X2APIC_IRR2	Thread	x2APIC Interrupt Request Register Bits [95:64] (R/0)
823H	2083	IA32_X2APIC_IRR3	Thread	x2APIC Interrupt Request Register Bits [127:96] (R/0)
824H	2084	IA32_X2APIC_IRR4	Thread	x2APIC Interrupt Request Register Bits [159:128] (R/O)
825H	2085	IA32_X2APIC_IRR5	Thread	x2APIC Interrupt Request Register Bits [191:160] (R/O)
826H	2086	IA32_X2APIC_IRR6	Thread	x2APIC Interrupt Request Register Bits [223:192] (R/O)
827H	2087	IA32_X2APIC_IRR7	Thread	x2APIC Interrupt Request Register Bits [255:224] (R/O)
828H	2088	IA32_X2APIC_ESR	Thread	x2APIC Error Status Register (R/W)
82FH	2095	IA32_X2APIC_LVT_CMCI	Thread	x2APIC LVT Corrected Machine Check Interrupt Register (R/W)
830H	2096	IA32_X2APIC_ICR	Thread	x2APIC Interrupt Command Register (R/W)
832H	2098	IA32_X2APIC_LVT_TIMER	Thread	x2APIC LVT Timer Interrupt Register (R/W)
833H	2099	IA32_X2APIC_LVT_THERMAL	Thread	x2APIC LVT Thermal Sensor Interrupt Register (R/W)
834H	2100	IA32_X2APIC_LVT_PMI	Thread	x2APIC LVT Performance Monitor Register (R/W)
835H	2101	IA32_X2APIC_LVT_LINT0	Thread	x2APIC LVT LINTO Register (R/W)
836H	2102	IA32_X2APIC_LVT_LINT1	Thread	x2APIC LVT LINT1 Register (R/W)
837H	2103	IA32_X2APIC_LVT_ERROR	Thread	x2APIC LVT Error Register (R/W)
838H	2104	IA32_X2APIC_INIT_COUNT	Thread	x2APIC Initial Count Register (R/W)
839H	2105	IA32_X2APIC_CUR_COUNT	Thread	x2APIC Current Count Register (R/O)
83EH	2110	IA32_X2APIC_DIV_CONF	Thread	x2APIC Divide Configuration Register (R/W)
83FH	2111	IA32_X2APIC_SELF_IPI	Thread	x2APIC Self IPI Register (W/O)

Table 2-15. MSRs in Processors Based on Nehalem Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
0080H		IA32_EFER	Thread	Extended Feature Enables See Table 2-2.
C000_ 0081H		IA32_STAR	Thread	System Call Target Address (R/W) See Table 2-2.
C000_ 0082H		IA32_LSTAR	Thread	IA-32e Mode System Call Target Address (R/W) See Table 2-2.
C000_ 0084H		IA32_FMASK	Thread	System Call Flag Mask (R/W) See Table 2-2.
C000_ 0100H		IA32_FS_BASE	Thread	Map of BASE Address of FS (R/W) See Table 2-2.
C000_ 0101H		IA32_GS_BASE	Thread	Map of BASE Address of GS (R/W) See Table 2-2.
C000_ 0102H		IA32_KERNEL_GS_BASE	Thread	Swap Target of BASE Address of GS (R/W) See Table 2-2.
C000_ 0103H		IA32_TSC_AUX	Thread	AUXILIARY TSC Signature (R/W) See Table 2-2 and Section 18.17.2, "IA32_TSC_AUX Register and RDTSCP Support."

2.8.1 Additional MSRs in the Intel® Xeon® Processor 5500 and 3400 Series

Intel Xeon Processor 5500 and 3400 series support additional model-specific registers listed in Table 2-16. These MSRs also apply to Intel Core i7 and i5 processor family CPUID signature with DisplayFamily_DisplayModel of 06_1AH, 06_1EH, and 06_1FH, see Table 2-1.

Table 2-16. Additional MSRs in Intel® Xeon® Processor 5500 and 3400 Series

Regi Addı		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
1ADH	429	MSR_TURBO_RATIO_LIMIT	Package	Actual maximum turbo frequency is multiplied by 133.33MHz.
				(Not available in model 06_2EH.)
		7:0		Maximum Turbo Ratio Limit 1C (R/O)
				Maximum Turbo mode ratio limit with 1 core active.
		15:8		Maximum Turbo Ratio Limit 2C (R/O)
				Maximum Turbo mode ratio limit with 2 cores active.
		23:16		Maximum Turbo Ratio Limit 3C (R/O)
				Maximum Turbo mode ratio limit with 3 cores active.
		31:24		Maximum Turbo Ratio Limit 4C (R/O)
				Maximum Turbo mode ratio limit with 4 cores active.
		63:32		Reserved
301H	769	MSR_GQ_SNOOP_MESF	Package	

Table 2-16. Additional MSRs in Intel® Xeon® Processor 5500 and 3400 Series (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	_		
		0		From M to S (R/W)
		1		From E to S (R/W)
		2		From S to S (R/W)
		3		From F to S (R/W)
		4		From M to I (R/W)
		5		From E to I (R/W)
		6		From S to I (R/W)
		7		From F to I (R/W)
		63:8		Reserved
391H	913	MSR_UNCORE_PERF_GLOBAL_CTRL	Package	See Section 20.3.1.2.1, "Uncore Performance Monitoring Management Facility."
392H	914	MSR_UNCORE_PERF_GLOBAL_STATUS	Package	See Section 20.3.1.2.1, "Uncore Performance Monitoring Management Facility."
393H	915	MSR_UNCORE_PERF_GLOBAL_OVF_CTRL	Package	See Section 20.3.1.2.1, "Uncore Performance Monitoring Management Facility."
394H	916	MSR_UNCORE_FIXED_CTR0	Package	See Section 20.3.1.2.1, "Uncore Performance Monitoring Management Facility."
395H	917	MSR_UNCORE_FIXED_CTR_CTRL	Package	See Section 20.3.1.2.1, "Uncore Performance Monitoring Management Facility."
396H	918	MSR_UNCORE_ADDR_OPCODE_MATCH	Package	See Section 20.3.1.2.3, "Uncore Address/Opcode Match MSR."
3B0H	960	MSR_UNCORE_PMCO	Package	See Section 20.3.1.2.2, "Uncore Performance Event Configuration Facility."
3B1H	961	MSR_UNCORE_PMC1	Package	See Section 20.3.1.2.2, "Uncore Performance Event Configuration Facility."
3B2H	962	MSR_UNCORE_PMC2	Package	See Section 20.3.1.2.2, "Uncore Performance Event Configuration Facility."
3B3H	963	MSR_UNCORE_PMC3	Package	See Section 20.3.1.2.2, "Uncore Performance Event Configuration Facility."
3B4H	964	MSR_UNCORE_PMC4	Package	See Section 20.3.1.2.2, "Uncore Performance Event Configuration Facility."
3B5H	965	MSR_UNCORE_PMC5	Package	See Section 20.3.1.2.2, "Uncore Performance Event Configuration Facility."
3B6H	966	MSR_UNCORE_PMC6	Package	See Section 20.3.1.2.2, "Uncore Performance Event Configuration Facility."
3B7H	967	MSR_UNCORE_PMC7	Package	See Section 20.3.1.2.2, "Uncore Performance Event Configuration Facility."
3C0H	944	MSR_UNCORE_PERFEVTSEL0	Package	See Section 20.3.1.2.2, "Uncore Performance Event Configuration Facility."
3C1H	945	MSR_UNCORE_PERFEVTSEL1	Package	See Section 20.3.1.2.2, "Uncore Performance Event Configuration Facility."

Table 2-16. Additional MSRs in Intel® Xeon® Processor 5500 and 3400 Series (Contd.)

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
3C2H	946	MSR_UNCORE_PERFEVTSEL2	Package	See Section 20.3.1.2.2, "Uncore Performance Event Configuration Facility."
3C3H	947	MSR_UNCORE_PERFEVTSEL3	Package	See Section 20.3.1.2.2, "Uncore Performance Event Configuration Facility."
3C4H	948	MSR_UNCORE_PERFEVTSEL4	Package	See Section 20.3.1.2.2, "Uncore Performance Event Configuration Facility."
3C5H	949	MSR_UNCORE_PERFEVTSEL5	Package	See Section 20.3.1.2.2, "Uncore Performance Event Configuration Facility."
3C6H	950	MSR_UNCORE_PERFEVTSEL6	Package	See Section 20.3.1.2.2, "Uncore Performance Event Configuration Facility."
3C7H	951	MSR_UNCORE_PERFEVTSEL7	Package	See Section 20.3.1.2.2, "Uncore Performance Event Configuration Facility."

2.8.2 Additional MSRs in the Intel® Xeon® Processor 7500 Series

Intel Xeon Processor 7500 series support MSRs listed in Table 2-15 (except MSR address 1ADH) and additional model-specific registers listed in Table 2-17. These processors have a CPUID signature with DisplayFamily DisplayModel of 06 2EH.

Table 2-17. Additional MSRs in Intel® Xeon® Processor 7500 Series

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
1ADH	429	MSR_TURBO_RATIO_LIMIT	Package	Reserved
				Attempt to read/write will cause #UD.
289H	649	IA32_MC9_CTL2	Package	See Table 2-2.
28AH	650	IA32_MC10_CTL2	Package	See Table 2-2.
28BH	651	IA32_MC11_CTL2	Package	See Table 2-2.
28CH	652	IA32_MC12_CTL2	Package	See Table 2-2.
28DH	653	IA32_MC13_CTL2	Package	See Table 2-2.
28EH	654	IA32_MC14_CTL2	Package	See Table 2-2.
28FH	655	IA32_MC15_CTL2	Package	See Table 2-2.
290H	656	IA32_MC16_CTL2	Package	See Table 2-2.
291H	657	IA32_MC17_CTL2	Package	See Table 2-2.
292H	658	IA32_MC18_CTL2	Package	See Table 2-2.
293H	659	IA32_MC19_CTL2	Package	See Table 2-2.
294H	660	IA32_MC20_CTL2	Package	See Table 2-2.
295H	661	IA32_MC21_CTL2	Package	See Table 2-2.
394H	816	MSR_W_PMON_FIXED_CTR	Package	Uncore W-box perfmon fixed counter.
395H	817	MSR_W_PMON_FIXED_CTR_CTL	Package	Uncore U-box perfmon fixed counter control MSR.
424H	1060	IA32_MC9_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."

Table 2-17. Additional MSRs in Intel® Xeon® Processor 7500 Series (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
425H	1061	IA32_MC9_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
426H	1062	IA32_MC9_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
427H	1063	IA32_MC9_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
428H	1064	IA32_MC10_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
429H	1065	IA32_MC10_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
42AH	1066	IA32_MC10_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
42BH	1067	IA32_MC10_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
42CH	1068	IA32_MC11_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
42DH	1069	IA32_MC11_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
42EH	1070	IA32_MC11_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
42FH	1071	IA32_MC11_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
430H	1072	IA32_MC12_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
431H	1073	IA32_MC12_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
432H	1074	IA32_MC12_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
433H	1075	IA32_MC12_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
434H	1076	IA32_MC13_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
435H	1077	IA32_MC13_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
436H	1078	IA32_MC13_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
437H	1079	IA32_MC13_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
438H	1080	IA32_MC14_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
439H	1081	IA32_MC14_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
43AH	1082	IA32_MC14_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
43BH	1083	IA32_MC14_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
43CH	1084	IA32_MC15_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
43DH	1085	IA32_MC15_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
43EH	1086	IA32_MC15_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
43FH	1087	IA32_MC15_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
440H	1088	IA32_MC16_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
441H	1089	IA32_MC16_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
442H	1090	IA32_MC16_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
443H	1091	IA32_MC16_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."

Table 2-17. Additional MSRs in Intel® Xeon® Processor 7500 Series (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
444H	1092	IA32_MC17_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
445H	1093	IA32_MC17_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
446H	1094	IA32_MC17_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
447H	1095	IA32_MC17_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
448H	1096	IA32_MC18_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
449H	1097	IA32_MC18_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
44AH	1098	IA32_MC18_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
44BH	1099	IA32_MC18_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
44CH	1100	IA32_MC19_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
44DH	1101	IA32_MC19_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
44EH	1102	IA32_MC19_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
44FH	1103	IA32_MC19_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
450H	1104	IA32_MC20_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
451H	1105	IA32_MC20_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
452H	1106	IA32_MC20_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
453H	1107	IA32_MC20_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
454H	1108	IA32_MC21_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
455H	1109	IA32_MC21_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
456H	1110	IA32_MC21_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
457H	1111	IA32_MC21_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
COOH	3072	MSR_U_PMON_GLOBAL_CTRL	Package	Uncore U-box perfmon global control MSR.
CO1H	3073	MSR_U_PMON_GLOBAL_STATUS	Package	Uncore U-box perfmon global status MSR.
C02H	3074	MSR_U_PMON_GLOBAL_OVF_CTRL	Package	Uncore U-box perfmon global overflow control MSR.
C10H	3088	MSR_U_PMON_EVNT_SEL	Package	Uncore U-box perfmon event select MSR.
C11H	3089	MSR_U_PMON_CTR	Package	Uncore U-box perfmon counter MSR.
C20H	3104	MSR_B0_PMON_BOX_CTRL	Package	Uncore B-box 0 perfmon local box control MSR.
C21H	3105	MSR_B0_PMON_BOX_STATUS	Package	Uncore B-box 0 perfmon local box status MSR.
C22H	3106	MSR_B0_PMON_BOX_OVF_CTRL	Package	Uncore B-box 0 perfmon local box overflow control MSR.
C30H	3120	MSR_B0_PMON_EVNT_SEL0	Package	Uncore B-box 0 perfmon event select MSR.

Table 2-17. Additional MSRs in Intel® Xeon® Processor 7500 Series (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
C31H	3121	MSR_B0_PMON_CTR0	Package	Uncore B-box 0 perfmon counter MSR.
C32H	3122	MSR_B0_PMON_EVNT_SEL1	Package	Uncore B-box 0 perfmon event select MSR.
C33H	3123	MSR_B0_PMON_CTR1	Package	Uncore B-box 0 perfmon counter MSR.
C34H	3124	MSR_B0_PMON_EVNT_SEL2	Package	Uncore B-box 0 perfmon event select MSR.
C35H	3125	MSR_B0_PMON_CTR2	Package	Uncore B-box 0 perfmon counter MSR.
C36H	3126	MSR_B0_PMON_EVNT_SEL3	Package	Uncore B-box 0 perfmon event select MSR.
C37H	3127	MSR_B0_PMON_CTR3	Package	Uncore B-box 0 perfmon counter MSR.
C40H	3136	MSR_SO_PMON_BOX_CTRL	Package	Uncore S-box 0 perfmon local box control MSR.
C41H	3137	MSR_SO_PMON_BOX_STATUS	Package	Uncore S-box 0 perfmon local box status MSR.
C42H	3138	MSR_SO_PMON_BOX_OVF_CTRL	Package	Uncore S-box 0 perfmon local box overflow control MSR.
C50H	3152	MSR_S0_PMON_EVNT_SEL0	Package	Uncore S-box 0 perfmon event select MSR.
C51H	3153	MSR_SO_PMON_CTRO	Package	Uncore S-box 0 perfmon counter MSR.
C52H	3154	MSR_S0_PMON_EVNT_SEL1	Package	Uncore S-box 0 perfmon event select MSR.
C53H	3155	MSR_S0_PMON_CTR1	Package	Uncore S-box 0 perfmon counter MSR.
C54H	3156	MSR_S0_PMON_EVNT_SEL2	Package	Uncore S-box 0 perfmon event select MSR.
C55H	3157	MSR_S0_PMON_CTR2	Package	Uncore S-box 0 perfmon counter MSR.
C56H	3158	MSR_SO_PMON_EVNT_SEL3	Package	Uncore S-box 0 perfmon event select MSR.
C57H	3159	MSR_S0_PMON_CTR3	Package	Uncore S-box 0 perfmon counter MSR.
C60H	3168	MSR_B1_PMON_BOX_CTRL	Package	Uncore B-box 1 perfmon local box control MSR.
C61H	3169	MSR_B1_PMON_BOX_STATUS	Package	Uncore B-box 1 perfmon local box status MSR.
C62H	3170	MSR_B1_PMON_BOX_OVF_CTRL	Package	Uncore B-box 1 perfmon local box overflow control MSR.
C70H	3184	MSR_B1_PMON_EVNT_SEL0	Package	Uncore B-box 1 perfmon event select MSR.
C71H	3185	MSR_B1_PMON_CTR0	Package	Uncore B-box 1 perfmon counter MSR.
C72H	3186	MSR_B1_PMON_EVNT_SEL1	Package	Uncore B-box 1 perfmon event select MSR.
C73H	3187	MSR_B1_PMON_CTR1	Package	Uncore B-box 1 perfmon counter MSR.

Table 2-17. Additional MSRs in Intel® Xeon® Processor 7500 Series (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
C74H	3188	MSR_B1_PMON_EVNT_SEL2	Package	Uncore B-box 1 perfmon event select MSR.
C75H	3189	MSR_B1_PMON_CTR2	Package	Uncore B-box 1 perfmon counter MSR.
C76H	3190	MSR_B1_PMON_EVNT_SEL3	Package	Uncore B-box 1vperfmon event select MSR.
C77H	3191	MSR_B1_PMON_CTR3	Package	Uncore B-box 1 perfmon counter MSR.
C80H	3120	MSR_W_PMON_BOX_CTRL	Package	Uncore W-box perfmon local box control MSR.
C81H	3121	MSR_W_PMON_BOX_STATUS	Package	Uncore W-box perfmon local box status MSR.
C82H	3122	MSR_W_PMON_BOX_OVF_CTRL	Package	Uncore W-box perfmon local box overflow control MSR.
C90H	3136	MSR_W_PMON_EVNT_SEL0	Package	Uncore W-box perfmon event select MSR.
C91H	3137	MSR_W_PMON_CTR0	Package	Uncore W-box perfmon counter MSR.
C92H	3138	MSR_W_PMON_EVNT_SEL1	Package	Uncore W-box perfmon event select MSR.
C93H	3139	MSR_W_PMON_CTR1	Package	Uncore W-box perfmon counter MSR.
C94H	3140	MSR_W_PMON_EVNT_SEL2	Package	Uncore W-box perfmon event select MSR.
C95H	3141	MSR_W_PMON_CTR2	Package	Uncore W-box perfmon counter MSR.
C96H	3142	MSR_W_PMON_EVNT_SEL3	Package	Uncore W-box perfmon event select MSR.
C97H	3143	MSR_W_PMON_CTR3	Package	Uncore W-box perfmon counter MSR.
CAOH	3232	MSR_MO_PMON_BOX_CTRL	Package	Uncore M-box 0 perfmon local box control MSR.
CA1H	3233	MSR_MO_PMON_BOX_STATUS	Package	Uncore M-box O perfmon local box status MSR.
CA2H	3234	MSR_MO_PMON_BOX_OVF_CTRL	Package	Uncore M-box 0 perfmon local box overflow control MSR.
CA4H	3236	MSR_MO_PMON_TIMESTAMP	Package	Uncore M-box 0 perfmon time stamp unit select MSR.
CA5H	3237	MSR_MO_PMON_DSP	Package	Uncore M-box 0 perfmon DSP unit select MSR.
CA6H	3238	MSR_MO_PMON_ISS	Package	Uncore M-box 0 perfmon ISS unit select MSR.
CA7H	3239	MSR_M0_PMON_MAP	Package	Uncore M-box 0 perfmon MAP unit select MSR.
CA8H	3240	MSR_MO_PMON_MSC_THR	Package	Uncore M-box 0 perfmon MIC THR select MSR.
CA9H	3241	MSR_M0_PMON_PGT	Package	Uncore M-box 0 perfmon PGT unit select MSR.
CAAH	3242	MSR_MO_PMON_PLD	Package	Uncore M-box 0 perfmon PLD unit select MSR.
CABH	3243	MSR_MO_PMON_ZDP	Package	Uncore M-box 0 perfmon ZDP unit select MSR.
СВОН	3248	MSR_MO_PMON_EVNT_SEL0	Package	Uncore M-box 0 perfmon event select MSR.
CB1H	3249	MSR_MO_PMON_CTRO	Package	Uncore M-box 0 perfmon counter MSR.

Table 2-17. Additional MSRs in Intel® Xeon® Processor 7500 Series (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
CB2H	3250	MSR_M0_PMON_EVNT_SEL1	Package	Uncore M-box 0 perfmon event select MSR.
СВЗН	3251	MSR_M0_PMON_CTR1	Package	Uncore M-box 0 perfmon counter MSR.
CB4H	3252	MSR_M0_PMON_EVNT_SEL2	Package	Uncore M-box 0 perfmon event select MSR.
CB5H	3253	MSR_M0_PMON_CTR2	Package	Uncore M-box 0 perfmon counter MSR.
СВ6Н	3254	MSR_MO_PMON_EVNT_SEL3	Package	Uncore M-box 0 perfmon event select MSR.
CB7H	3255	MSR_MO_PMON_CTR3	Package	Uncore M-box 0 perfmon counter MSR.
CB8H	3256	MSR_M0_PMON_EVNT_SEL4	Package	Uncore M-box 0 perfmon event select MSR.
CB9H	3257	MSR_M0_PMON_CTR4	Package	Uncore M-box 0 perfmon counter MSR.
CBAH	3258	MSR_M0_PMON_EVNT_SEL5	Package	Uncore M-box 0 perfmon event select MSR.
CBBH	3259	MSR_MO_PMON_CTR5	Package	Uncore M-box 0 perfmon counter MSR.
CCOH	3264	MSR_S1_PMON_BOX_CTRL	Package	Uncore S-box 1 perfmon local box control MSR.
CC1H	3265	MSR_S1_PMON_BOX_STATUS	Package	Uncore S-box 1 perfmon local box status MSR.
CC2H	3266	MSR_S1_PMON_BOX_OVF_CTRL	Package	Uncore S-box 1 perfmon local box overflow control MSR.
CDOH	3280	MSR_S1_PMON_EVNT_SEL0	Package	Uncore S-box 1 perfmon event select MSR.
CD1H	3281	MSR_S1_PMON_CTR0	Package	Uncore S-box 1 perfmon counter MSR.
CD2H	3282	MSR_S1_PMON_EVNT_SEL1	Package	Uncore S-box 1 perfmon event select MSR.
CD3H	3283	MSR_S1_PMON_CTR1	Package	Uncore S-box 1 perfmon counter MSR.
CD4H	3284	MSR_S1_PMON_EVNT_SEL2	Package	Uncore S-box 1 perfmon event select MSR.
CD5H	3285	MSR_S1_PMON_CTR2	Package	Uncore S-box 1 perfmon counter MSR.
CD6H	3286	MSR_S1_PMON_EVNT_SEL3	Package	Uncore S-box 1 perfmon event select MSR.
CD7H	3287	MSR_S1_PMON_CTR3	Package	Uncore S-box 1 perfmon counter MSR.
CEOH	3296	MSR_M1_PMON_BOX_CTRL	Package	Uncore M-box 1 perfmon local box control MSR.
CE1H	3297	MSR_M1_PMON_BOX_STATUS	Package	Uncore M-box 1 perfmon local box status MSR.
CE2H	3298	MSR_M1_PMON_BOX_OVF_CTRL	Package	Uncore M-box 1 perfmon local box overflow control MSR.
CE4H	3300	MSR_M1_PMON_TIMESTAMP	Package	Uncore M-box 1 perfmon time stamp unit select MSR.
CE5H	3301	MSR_M1_PMON_DSP	Package	Uncore M-box 1 perfmon DSP unit select MSR.

Table 2-17. Additional MSRs in Intel® Xeon® Processor 7500 Series (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
CE6H	3302	MSR_M1_PMON_ISS	Package	Uncore M-box 1 perfmon ISS unit select MSR.
CE7H	3303	MSR_M1_PMON_MAP	Package	Uncore M-box 1 perfmon MAP unit select MSR.
CE8H	3304	MSR_M1_PMON_MSC_THR	Package	Uncore M-box 1 perfmon MIC THR select MSR.
CE9H	3305	MSR_M1_PMON_PGT	Package	Uncore M-box 1 perfmon PGT unit select MSR.
CEAH	3306	MSR_M1_PMON_PLD	Package	Uncore M-box 1 perfmon PLD unit select MSR.
CEBH	3307	MSR_M1_PMON_ZDP	Package	Uncore M-box 1 perfmon ZDP unit select MSR.
CF0H	3312	MSR_M1_PMON_EVNT_SEL0	Package	Uncore M-box 1 perfmon event select MSR.
CF1H	3313	MSR_M1_PMON_CTR0	Package	Uncore M-box 1 perfmon counter MSR.
CF2H	3314	MSR_M1_PMON_EVNT_SEL1	Package	Uncore M-box 1 perfmon event select MSR.
CF3H	3315	MSR_M1_PMON_CTR1	Package	Uncore M-box 1 perfmon counter MSR.
CF4H	3316	MSR_M1_PMON_EVNT_SEL2	Package	Uncore M-box 1 perfmon event select MSR.
CF5H	3317	MSR_M1_PMON_CTR2	Package	Uncore M-box 1 perfmon counter MSR.
CF6H	3318	MSR_M1_PMON_EVNT_SEL3	Package	Uncore M-box 1 perfmon event select MSR.
CF7H	3319	MSR_M1_PMON_CTR3	Package	Uncore M-box 1 perfmon counter MSR.
CF8H	3320	MSR_M1_PMON_EVNT_SEL4	Package	Uncore M-box 1 perfmon event select MSR.
CF9H	3321	MSR_M1_PMON_CTR4	Package	Uncore M-box 1 perfmon counter MSR.
CFAH	3322	MSR_M1_PMON_EVNT_SEL5	Package	Uncore M-box 1 perfmon event select MSR.
CFBH	3323	MSR_M1_PMON_CTR5	Package	Uncore M-box 1 perfmon counter MSR.
D00H	3328	MSR_CO_PMON_BOX_CTRL	Package	Uncore C-box 0 perfmon local box control MSR.
D01H	3329	MSR_CO_PMON_BOX_STATUS	Package	Uncore C-box O perfmon local box status MSR.
D02H	3330	MSR_CO_PMON_BOX_OVF_CTRL	Package	Uncore C-box O perfmon local box overflow control MSR.
D10H	3344	MSR_CO_PMON_EVNT_SELO	Package	Uncore C-box 0 perfmon event select MSR.
D11H	3345	MSR_CO_PMON_CTRO	Package	Uncore C-box O perfmon counter MSR.
D12H	3346	MSR_CO_PMON_EVNT_SEL1	Package	Uncore C-box 0 perfmon event select MSR.
D13H	3347	MSR_CO_PMON_CTR1	Package	Uncore C-box O perfmon counter MSR.
D14H	3348	MSR_CO_PMON_EVNT_SEL2	Package	Uncore C-box 0 perfmon event select MSR.
D15H	3349	MSR_CO_PMON_CTR2	Package	Uncore C-box 0 perfmon counter MSR.

Table 2-17. Additional MSRs in Intel® Xeon® Processor 7500 Series (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
D16H	3350	MSR_CO_PMON_EVNT_SEL3	Package	Uncore C-box 0 perfmon event select MSR.
D17H	3351	MSR_CO_PMON_CTR3	Package	Uncore C-box 0 perfmon counter MSR.
D18H	3352	MSR_CO_PMON_EVNT_SEL4	Package	Uncore C-box 0 perfmon event select MSR.
D19H	3353	MSR_CO_PMON_CTR4	Package	Uncore C-box 0 perfmon counter MSR.
D1AH	3354	MSR_CO_PMON_EVNT_SEL5	Package	Uncore C-box 0 perfmon event select MSR.
D1BH	3355	MSR_CO_PMON_CTR5	Package	Uncore C-box O perfmon counter MSR.
D20H	3360	MSR_C4_PMON_BOX_CTRL	Package	Uncore C-box 4 perfmon local box control MSR.
D21H	3361	MSR_C4_PMON_BOX_STATUS	Package	Uncore C-box 4 perfmon local box status MSR.
D22H	3362	MSR_C4_PMON_BOX_OVF_CTRL	Package	Uncore C-box 4 perfmon local box overflow control MSR.
D30H	3376	MSR_C4_PMON_EVNT_SEL0	Package	Uncore C-box 4 perfmon event select MSR.
D31H	3377	MSR_C4_PMON_CTR0	Package	Uncore C-box 4 perfmon counter MSR.
D32H	3378	MSR_C4_PMON_EVNT_SEL1	Package	Uncore C-box 4 perfmon event select MSR.
D33H	3379	MSR_C4_PMON_CTR1	Package	Uncore C-box 4 perfmon counter MSR.
D34H	3380	MSR_C4_PMON_EVNT_SEL2	Package	Uncore C-box 4 perfmon event select MSR.
D35H	3381	MSR_C4_PMON_CTR2	Package	Uncore C-box 4 perfmon counter MSR.
D36H	3382	MSR_C4_PMON_EVNT_SEL3	Package	Uncore C-box 4 perfmon event select MSR.
D37H	3383	MSR_C4_PMON_CTR3	Package	Uncore C-box 4 perfmon counter MSR.
D38H	3384	MSR_C4_PMON_EVNT_SEL4	Package	Uncore C-box 4 perfmon event select MSR.
D39H	3385	MSR_C4_PMON_CTR4	Package	Uncore C-box 4 perfmon counter MSR.
D3AH	3386	MSR_C4_PMON_EVNT_SEL5	Package	Uncore C-box 4 perfmon event select MSR.
D3BH	3387	MSR_C4_PMON_CTR5	Package	Uncore C-box 4 perfmon counter MSR.
D40H	3392	MSR_C2_PMON_BOX_CTRL	Package	Uncore C-box 2 perfmon local box control MSR.
D41H	3393	MSR_C2_PMON_BOX_STATUS	Package	Uncore C-box 2 perfmon local box status MSR.
D42H	3394	MSR_C2_PMON_BOX_OVF_CTRL	Package	Uncore C-box 2 perfmon local box overflow control MSR.
D50H	3408	MSR_C2_PMON_EVNT_SEL0	Package	Uncore C-box 2 perfmon event select MSR.
D51H	3409	MSR_C2_PMON_CTR0	Package	Uncore C-box 2 perfmon counter MSR.

Table 2-17. Additional MSRs in Intel® Xeon® Processor 7500 Series (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
D52H	3410	MSR_C2_PMON_EVNT_SEL1	Package	Uncore C-box 2 perfmon event select MSR.
D53H	3411	MSR_C2_PMON_CTR1	Package	Uncore C-box 2 perfmon counter MSR.
D54H	3412	MSR_C2_PMON_EVNT_SEL2	Package	Uncore C-box 2 perfmon event select MSR.
D55H	3413	MSR_C2_PMON_CTR2	Package	Uncore C-box 2 perfmon counter MSR.
D56H	3414	MSR_C2_PMON_EVNT_SEL3	Package	Uncore C-box 2 perfmon event select MSR.
D57H	3415	MSR_C2_PMON_CTR3	Package	Uncore C-box 2 perfmon counter MSR.
D58H	3416	MSR_C2_PMON_EVNT_SEL4	Package	Uncore C-box 2 perfmon event select MSR.
D59H	3417	MSR_C2_PMON_CTR4	Package	Uncore C-box 2 perfmon counter MSR.
D5AH	3418	MSR_C2_PMON_EVNT_SEL5	Package	Uncore C-box 2 perfmon event select MSR.
D5BH	3419	MSR_C2_PMON_CTR5	Package	Uncore C-box 2 perfmon counter MSR.
D60H	3424	MSR_C6_PMON_BOX_CTRL	Package	Uncore C-box 6 perfmon local box control MSR.
D61H	3425	MSR_C6_PMON_BOX_STATUS	Package	Uncore C-box 6 perfmon local box status MSR.
D62H	3426	MSR_C6_PMON_BOX_OVF_CTRL	Package	Uncore C-box 6 perfmon local box overflow control MSR.
D70H	3440	MSR_C6_PMON_EVNT_SEL0	Package	Uncore C-box 6 perfmon event select MSR.
D71H	3441	MSR_C6_PMON_CTR0	Package	Uncore C-box 6 perfmon counter MSR.
D72H	3442	MSR_C6_PMON_EVNT_SEL1	Package	Uncore C-box 6 perfmon event select MSR.
D73H	3443	MSR_C6_PMON_CTR1	Package	Uncore C-box 6 perfmon counter MSR.
D74H	3444	MSR_C6_PMON_EVNT_SEL2	Package	Uncore C-box 6 perfmon event select MSR.
D75H	3445	MSR_C6_PMON_CTR2	Package	Uncore C-box 6 perfmon counter MSR.
D76H	3446	MSR_C6_PMON_EVNT_SEL3	Package	Uncore C-box 6 perfmon event select MSR.
D77H	3447	MSR_C6_PMON_CTR3	Package	Uncore C-box 6 perfmon counter MSR.
D78H	3448	MSR_C6_PMON_EVNT_SEL4	Package	Uncore C-box 6 perfmon event select MSR.
D79H	3449	MSR_C6_PMON_CTR4	Package	Uncore C-box 6 perfmon counter MSR.
D7AH	3450	MSR_C6_PMON_EVNT_SEL5	Package	Uncore C-box 6 perfmon event select MSR.
D7BH	3451	MSR_C6_PMON_CTR5	Package	Uncore C-box 6 perfmon counter MSR.

Table 2-17. Additional MSRs in Intel® Xeon® Processor 7500 Series (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
D80H	3456	MSR_C1_PMON_BOX_CTRL	Package	Uncore C-box 1 perfmon local box control MSR.
D81H	3457	MSR_C1_PMON_BOX_STATUS	Package	Uncore C-box 1 perfmon local box status MSR.
D82H	3458	MSR_C1_PMON_BOX_OVF_CTRL	Package	Uncore C-box 1 perfmon local box overflow control MSR.
D90H	3472	MSR_C1_PMON_EVNT_SEL0	Package	Uncore C-box 1 perfmon event select MSR.
D91H	3473	MSR_C1_PMON_CTR0	Package	Uncore C-box 1 perfmon counter MSR.
D92H	3474	MSR_C1_PMON_EVNT_SEL1	Package	Uncore C-box 1 perfmon event select MSR.
D93H	3475	MSR_C1_PMON_CTR1	Package	Uncore C-box 1 perfmon counter MSR.
D94H	3476	MSR_C1_PMON_EVNT_SEL2	Package	Uncore C-box 1 perfmon event select MSR.
D95H	3477	MSR_C1_PMON_CTR2	Package	Uncore C-box 1 perfmon counter MSR.
D96H	3478	MSR_C1_PMON_EVNT_SEL3	Package	Uncore C-box 1 perfmon event select MSR.
D97H	3479	MSR_C1_PMON_CTR3	Package	Uncore C-box 1 perfmon counter MSR.
D98H	3480	MSR_C1_PMON_EVNT_SEL4	Package	Uncore C-box 1 perfmon event select MSR.
D99H	3481	MSR_C1_PMON_CTR4	Package	Uncore C-box 1 perfmon counter MSR.
D9AH	3482	MSR_C1_PMON_EVNT_SEL5	Package	Uncore C-box 1 perfmon event select MSR.
D9BH	3483	MSR_C1_PMON_CTR5	Package	Uncore C-box 1 perfmon counter MSR.
DAOH	3488	MSR_C5_PMON_BOX_CTRL	Package	Uncore C-box 5 perfmon local box control MSR.
DA1H	3489	MSR_C5_PMON_BOX_STATUS	Package	Uncore C-box 5 perfmon local box status MSR.
DA2H	3490	MSR_C5_PMON_BOX_OVF_CTRL	Package	Uncore C-box 5 perfmon local box overflow control MSR.
DB0H	3504	MSR_C5_PMON_EVNT_SEL0	Package	Uncore C-box 5 perfmon event select MSR.
DB1H	3505	MSR_C5_PMON_CTR0	Package	Uncore C-box 5 perfmon counter MSR.
DB2H	3506	MSR_C5_PMON_EVNT_SEL1	Package	Uncore C-box 5 perfmon event select MSR.
DB3H	3507	MSR_C5_PMON_CTR1	Package	Uncore C-box 5 perfmon counter MSR.
DB4H	3508	MSR_C5_PMON_EVNT_SEL2	Package	Uncore C-box 5 perfmon event select MSR.
DB5H	3509	MSR_C5_PMON_CTR2	Package	Uncore C-box 5 perfmon counter MSR.
DB6H	3510	MSR_C5_PMON_EVNT_SEL3	Package	Uncore C-box 5 perfmon event select MSR.
DB7H	3511	MSR_C5_PMON_CTR3	Package	Uncore C-box 5 perfmon counter MSR.

Table 2-17. Additional MSRs in Intel® Xeon® Processor 7500 Series (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
DB8H	3512	MSR_C5_PMON_EVNT_SEL4	Package	Uncore C-box 5 perfmon event select MSR.
DB9H	3513	MSR_C5_PMON_CTR4	Package	Uncore C-box 5 perfmon counter MSR.
DBAH	3514	MSR_C5_PMON_EVNT_SEL5	Package	Uncore C-box 5 perfmon event select MSR.
DBBH	3515	MSR_C5_PMON_CTR5	Package	Uncore C-box 5 perfmon counter MSR.
DCOH	3520	MSR_C3_PMON_BOX_CTRL	Package	Uncore C-box 3 perfmon local box control MSR.
DC1H	3521	MSR_C3_PMON_BOX_STATUS	Package	Uncore C-box 3 perfmon local box status MSR.
DC2H	3522	MSR_C3_PMON_BOX_OVF_CTRL	Package	Uncore C-box 3 perfmon local box overflow control MSR.
DDOH	3536	MSR_C3_PMON_EVNT_SEL0	Package	Uncore C-box 3 perfmon event select MSR.
DD1H	3537	MSR_C3_PMON_CTR0	Package	Uncore C-box 3 perfmon counter MSR.
DD2H	3538	MSR_C3_PMON_EVNT_SEL1	Package	Uncore C-box 3 perfmon event select MSR.
DD3H	3539	MSR_C3_PMON_CTR1	Package	Uncore C-box 3 perfmon counter MSR.
DD4H	3540	MSR_C3_PMON_EVNT_SEL2	Package	Uncore C-box 3 perfmon event select MSR.
DD5H	3541	MSR_C3_PMON_CTR2	Package	Uncore C-box 3 perfmon counter MSR.
DD6H	3542	MSR_C3_PMON_EVNT_SEL3	Package	Uncore C-box 3 perfmon event select MSR.
DD7H	3543	MSR_C3_PMON_CTR3	Package	Uncore C-box 3 perfmon counter MSR.
DD8H	3544	MSR_C3_PMON_EVNT_SEL4	Package	Uncore C-box 3 perfmon event select MSR.
DD9H	3545	MSR_C3_PMON_CTR4	Package	Uncore C-box 3 perfmon counter MSR.
DDAH	3546	MSR_C3_PMON_EVNT_SEL5	Package	Uncore C-box 3 perfmon event select MSR.
DDBH	3547	MSR_C3_PMON_CTR5	Package	Uncore C-box 3 perfmon counter MSR.
DEOH	3552	MSR_C7_PMON_BOX_CTRL	Package	Uncore C-box 7 perfmon local box control MSR.
DE1H	3553	MSR_C7_PMON_BOX_STATUS	Package	Uncore C-box 7 perfmon local box status MSR.
DE2H	3554	MSR_C7_PMON_BOX_OVF_CTRL	Package	Uncore C-box 7 perfmon local box overflow control MSR.
DFOH	3568	MSR_C7_PMON_EVNT_SEL0	Package	Uncore C-box 7 perfmon event select MSR.
DF1H	3569	MSR_C7_PMON_CTR0	Package	Uncore C-box 7 perfmon counter MSR.
DF2H	3570	MSR_C7_PMON_EVNT_SEL1	Package	Uncore C-box 7 perfmon event select MSR.

Table 2-17. Additional MSRs in Intel® Xeon® Processor 7500 Series (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
DF3H	3571	MSR_C7_PMON_CTR1	Package	Uncore C-box 7 perfmon counter MSR.
DF4H	3572	MSR_C7_PMON_EVNT_SEL2	Package	Uncore C-box 7 perfmon event select MSR.
DF5H	3573	MSR_C7_PMON_CTR2	Package	Uncore C-box 7 perfmon counter MSR.
DF6H	3574	MSR_C7_PMON_EVNT_SEL3	Package	Uncore C-box 7 perfmon event select MSR.
DF7H	3575	MSR_C7_PMON_CTR3	Package	Uncore C-box 7 perfmon counter MSR.
DF8H	3576	MSR_C7_PMON_EVNT_SEL4	Package	Uncore C-box 7 perfmon event select MSR.
DF9H	3577	MSR_C7_PMON_CTR4	Package	Uncore C-box 7 perfmon counter MSR.
DFAH	3578	MSR_C7_PMON_EVNT_SEL5	Package	Uncore C-box 7 perfmon event select MSR.
DFBH	3579	MSR_C7_PMON_CTR5	Package	Uncore C-box 7 perfmon counter MSR.
EOOH	3584	MSR_RO_PMON_BOX_CTRL	Package	Uncore R-box 0 perfmon local box control MSR.
E01H	3585	MSR_R0_PMON_BOX_STATUS	Package	Uncore R-box 0 perfmon local box status MSR.
E02H	3586	MSR_R0_PMON_BOX_OVF_CTRL	Package	Uncore R-box 0 perfmon local box overflow control MSR.
E04H	3588	MSR_R0_PMON_IPERF0_P0	Package	Uncore R-box 0 perfmon IPERF0 unit Port 0 select MSR.
E05H	3589	MSR_R0_PMON_IPERF0_P1	Package	Uncore R-box 0 perfmon IPERF0 unit Port 1 select MSR.
E06H	3590	MSR_R0_PMON_IPERF0_P2	Package	Uncore R-box 0 perfmon IPERF0 unit Port 2 select MSR.
E07H	3591	MSR_R0_PMON_IPERF0_P3	Package	Uncore R-box 0 perfmon IPERF0 unit Port 3 select MSR.
E08H	3592	MSR_R0_PMON_IPERF0_P4	Package	Uncore R-box 0 perfmon IPERF0 unit Port 4 select MSR.
E09H	3593	MSR_R0_PMON_IPERF0_P5	Package	Uncore R-box 0 perfmon IPERF0 unit Port 5 select MSR.
EOAH	3594	MSR_R0_PMON_IPERF0_P6	Package	Uncore R-box 0 perfmon IPERF0 unit Port 6 select MSR.
EOBH	3595	MSR_R0_PMON_IPERF0_P7	Package	Uncore R-box 0 perfmon IPERF0 unit Port 7 select MSR.
EOCH	3596	MSR_R0_PMON_QLX_P0	Package	Uncore R-box 0 perfmon QLX unit Port 0 select MSR.
EODH	3597	MSR_R0_PMON_QLX_P1	Package	Uncore R-box 0 perfmon QLX unit Port 1 select MSR.
EOEH	3598	MSR_R0_PMON_QLX_P2	Package	Uncore R-box 0 perfmon QLX unit Port 2 select MSR.
EOFH	3599	MSR_R0_PMON_QLX_P3	Package	Uncore R-box 0 perfmon QLX unit Port 3 select MSR.
E10H	3600	MSR_R0_PMON_EVNT_SEL0	Package	Uncore R-box 0 perfmon event select MSR.
E11H	3601	MSR_R0_PMON_CTR0	Package	Uncore R-box 0 perfmon counter MSR.

Table 2-17. Additional MSRs in Intel® Xeon® Processor 7500 Series (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
E12H	3602	MSR_R0_PMON_EVNT_SEL1	Package	Uncore R-box 0 perfmon event select MSR.
E13H	3603	MSR_R0_PMON_CTR1	Package	Uncore R-box 0 perfmon counter MSR.
E14H	3604	MSR_R0_PMON_EVNT_SEL2	Package	Uncore R-box 0 perfmon event select MSR.
E15H	3605	MSR_R0_PMON_CTR2	Package	Uncore R-box O perfmon counter MSR.
E16H	3606	MSR_RO_PMON_EVNT_SEL3	Package	Uncore R-box 0 perfmon event select MSR.
E17H	3607	MSR_R0_PMON_CTR3	Package	Uncore R-box 0 perfmon counter MSR.
E18H	3608	MSR_R0_PMON_EVNT_SEL4	Package	Uncore R-box 0 perfmon event select MSR.
E19H	3609	MSR_R0_PMON_CTR4	Package	Uncore R-box 0 perfmon counter MSR.
E1AH	3610	MSR_R0_PMON_EVNT_SEL5	Package	Uncore R-box 0 perfmon event select MSR.
E1BH	3611	MSR_R0_PMON_CTR5	Package	Uncore R-box 0 perfmon counter MSR.
E1CH	3612	MSR_R0_PMON_EVNT_SEL6	Package	Uncore R-box 0 perfmon event select MSR.
E1DH	3613	MSR_R0_PMON_CTR6	Package	Uncore R-box O perfmon counter MSR.
E1EH	3614	MSR_R0_PMON_EVNT_SEL7	Package	Uncore R-box 0 perfmon event select MSR.
E1FH	3615	MSR_R0_PMON_CTR7	Package	Uncore R-box O perfmon counter MSR.
E20H	3616	MSR_R1_PMON_BOX_CTRL	Package	Uncore R-box 1 perfmon local box control MSR.
E21H	3617	MSR_R1_PMON_BOX_STATUS	Package	Uncore R-box 1 perfmon local box status MSR.
E22H	3618	MSR_R1_PMON_BOX_OVF_CTRL	Package	Uncore R-box 1 perfmon local box overflow control MSR.
E24H	3620	MSR_R1_PMON_IPERF1_P8	Package	Uncore R-box 1 perfmon IPERF1 unit Port 8 select MSR.
E25H	3621	MSR_R1_PMON_IPERF1_P9	Package	Uncore R-box 1 perfmon IPERF1 unit Port 9 select MSR.
E26H	3622	MSR_R1_PMON_IPERF1_P10	Package	Uncore R-box 1 perfmon IPERF1 unit Port 10 select MSR.
E27H	3623	MSR_R1_PMON_IPERF1_P11	Package	Uncore R-box 1 perfmon IPERF1 unit Port 11 select MSR.
E28H	3624	MSR_R1_PMON_IPERF1_P12	Package	Uncore R-box 1 perfmon IPERF1 unit Port 12 select MSR.
E29H	3625	MSR_R1_PMON_IPERF1_P13	Package	Uncore R-box 1 perfmon IPERF1 unit Port 13 select MSR.
E2AH	3626	MSR_R1_PMON_IPERF1_P14	Package	Uncore R-box 1 perfmon IPERF1 unit Port 14 select MSR.

Table 2-17. Additional MSRs in Intel® Xeon® Processor 7500 Series (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
E2BH	3627	MSR_R1_PMON_IPERF1_P15	Package	Uncore R-box 1 perfmon IPERF1 unit Port 15 select MSR.
E2CH	3628	MSR_R1_PMON_QLX_P4	Package	Uncore R-box 1 perfmon QLX unit Port 4 select MSR.
E2DH	3629	MSR_R1_PMON_QLX_P5	Package	Uncore R-box 1 perfmon QLX unit Port 5 select MSR.
E2EH	3630	MSR_R1_PMON_QLX_P6	Package	Uncore R-box 1 perfmon QLX unit Port 6 select MSR.
E2FH	3631	MSR_R1_PMON_QLX_P7	Package	Uncore R-box 1 perfmon QLX unit Port 7 select MSR.
E30H	3632	MSR_R1_PMON_EVNT_SEL8	Package	Uncore R-box 1 perfmon event select MSR.
E31H	3633	MSR_R1_PMON_CTR8	Package	Uncore R-box 1 perfmon counter MSR.
E32H	3634	MSR_R1_PMON_EVNT_SEL9	Package	Uncore R-box 1 perfmon event select MSR.
E33H	3635	MSR_R1_PMON_CTR9	Package	Uncore R-box 1 perfmon counter MSR.
E34H	3636	MSR_R1_PMON_EVNT_SEL10	Package	Uncore R-box 1 perfmon event select MSR.
E35H	3637	MSR_R1_PMON_CTR10	Package	Uncore R-box 1 perfmon counter MSR.
E36H	3638	MSR_R1_PMON_EVNT_SEL11	Package	Uncore R-box 1 perfmon event select MSR.
E37H	3639	MSR_R1_PMON_CTR11	Package	Uncore R-box 1 perfmon counter MSR.
E38H	3640	MSR_R1_PMON_EVNT_SEL12	Package	Uncore R-box 1 perfmon event select MSR.
E39H	3641	MSR_R1_PMON_CTR12	Package	Uncore R-box 1 perfmon counter MSR.
ЕЗАН	3642	MSR_R1_PMON_EVNT_SEL13	Package	Uncore R-box 1 perfmon event select MSR.
E3BH	3643	MSR_R1_PMON_CTR13	Package	Uncore R-box 1perfmon counter MSR.
E3CH	3644	MSR_R1_PMON_EVNT_SEL14	Package	Uncore R-box 1 perfmon event select MSR.
E3DH	3645	MSR_R1_PMON_CTR14	Package	Uncore R-box 1 perfmon counter MSR.
E3EH	3646	MSR_R1_PMON_EVNT_SEL15	Package	Uncore R-box 1 perfmon event select MSR.
E3FH	3647	MSR_R1_PMON_CTR15	Package	Uncore R-box 1 perfmon counter MSR.
E45H	3653	MSR_B0_PMON_MATCH	Package	Uncore B-box 0 perfmon local box match MSR.
E46H	3654	MSR_B0_PMON_MASK	Package	Uncore B-box 0 perfmon local box mask MSR.
E49H	3657	MSR_SO_PMON_MATCH	Package	Uncore S-box O perfmon local box match MSR.
E4AH	3658	MSR_SO_PMON_MASK	Package	Uncore S-box 0 perfmon local box mask MSR.
E4DH	3661	MSR_B1_PMON_MATCH	Package	Uncore B-box 1 perfmon local box match MSR.
E4EH	3662	MSR_B1_PMON_MASK	Package	Uncore B-box 1 perfmon local box mask MSR.
E54H	3668	MSR_MO_PMON_MM_CONFIG	Package	Uncore M-box 0 perfmon local box address match/mask config MSR.

Table 2-17. Additional MSRs in Intel® Xeon® Processor 7500 Series (Contd.)

Regi Add	ster ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
E55H	3669	MSR_MO_PMON_ADDR_MATCH	Package	Uncore M-box O perfmon local box address match MSR.
E56H	3670	MSR_MO_PMON_ADDR_MASK	Package	Uncore M-box O perfmon local box address mask MSR.
E59H	3673	MSR_S1_PMON_MATCH	Package	Uncore S-box 1 perfmon local box match MSR.
E5AH	3674	MSR_S1_PMON_MASK	Package	Uncore S-box 1 perfmon local box mask MSR.
E5CH	3676	MSR_M1_PMON_MM_CONFIG	Package	Uncore M-box 1 perfmon local box address match/mask config MSR.
E5DH	3677	MSR_M1_PMON_ADDR_MATCH	Package	Uncore M-box 1 perfmon local box address match MSR.
E5EH	3678	MSR_M1_PMON_ADDR_MASK	Package	Uncore M-box 1 perfmon local box address mask MSR.
3B5H	965	MSR_UNCORE_PMC5	Package	See Section 20.3.1.2.2, "Uncore Performance Event Configuration Facility."

2.9 MSRS IN THE INTEL® XEON® PROCESSOR 5600 SERIES (BASED ON WESTMERE MICROARCHITECTURE)

The Intel $^{\$}$ Xeon $^{\$}$ Processor 5600 Series (based on Westmere microarchitecture) supports the MSR interfaces listed in Table 2-15, Table 2-16, plus additional MSR listed in Table 2-18. These MSRs apply to Intel Core i7, i5, and i3 processor family with CPUID signature DisplayFamily_DisplayModel of 06_25H and 06_2CH, see Table 2-1.

Table 2-18. Additional MSRs Supported by Intel Processors Based on Westmere Microarchitecture

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
13CH	316	MSR_FEATURE_CONFIG	Core	AES Configuration (RW-L)
				Privileged post-BIOS agent must provide a #GP handler to handle unsuccessful read of this MSR.
		1:0		AES Configuration (RW-L)
				Upon a successful read of this MSR, the configuration of AES instruction set availability is as follows:
				11b: AES instructions are not available until next RESET.
				Otherwise, AES instructions are available.
				Note, AES instruction set is not available if read is unsuccessful. If the configuration is not 01b, AES instructions can be mis-configured if a privileged agent unintentionally writes 11b.
		63:2		Reserved
1A7H	423	MSR_OFFCORE_RSP_1	Thread	Offcore Response Event Select Register (R/W)

Table 2-18. Additional MSRs Supported by Intel Processors Based on Westmere Microarchitecture

Regi Addı		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
1ADH	429	MSR_TURBO_RATIO_LIMIT	Package	Maximum Ratio Limit of Turbo Mode
				R/O if MSR_PLATFORM_INFO.[28] = 0.
				R/W if MSR_PLATFORM_INFO.[28] = 1.
		7:0	Package	Maximum Ratio Limit for 1C
				Maximum turbo ratio limit of 1 core active.
		15:8	Package	Maximum Ratio Limit for 2C
				Maximum turbo ratio limit of 2 core active.
		23:16	Package	Maximum Ratio Limit for 3C
				Maximum turbo ratio limit of 3 core active.
		31:24	Package	Maximum Ratio Limit for 4C
				Maximum turbo ratio limit of 4 core active.
		39:32	Package	Maximum Ratio Limit for 5C
				Maximum turbo ratio limit of 5 core active.
		47:40	Package	Maximum Ratio Limit for 6C
				Maximum turbo ratio limit of 6 core active.
		63:48		Reserved
1B0H	432	IA32_ENERGY_PERF_BIAS	Package	See Table 2-2.

2.10 MSRS IN THE INTEL® XEON® PROCESSOR E7 FAMILY (BASED ON WESTMERE MICROARCHITECTURE)

The Intel $^{\$}$ Xeon $^{\$}$ Processor E7 Family (based on Westmere microarchitecture) supports the MSR interfaces listed in Table 2-15 (except MSR address 1ADH), Table 2-16, plus additional MSRs listed in Table 2-19. These processors have a CPUID signature with DisplayFamily_DisplayModel of 06_2FH.

Table 2-19. Additional MSRs Supported by Intel® Xeon® Processor E7 Family

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
13CH	316	MSR_FEATURE_CONFIG	Соге	AES Configuration (RW-L)
				Privileged post-BIOS agent must provide a #GP handler to handle unsuccessful read of this MSR.
		1:0		AES Configuration (RW-L)
				Upon a successful read of this MSR, the configuration of AES instruction set availability is as follows:
				11b: AES instructions are not available until next RESET.
				Otherwise, AES instructions are available.
				Note, AES instruction set is not available if read is unsuccessful. If the configuration is not 01b, AES instructions can be mis-configured if a privileged agent unintentionally writes 11b.

Table 2-19. Additional MSRs Supported by Intel® Xeon® Processor E7 Family (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		63:2		Reserved
1A7H	423	MSR_OFFCORE_RSP_1	Thread	Offcore Response Event Select Register (R/W)
1ADH	429	MSR_TURBO_RATIO_LIMIT	Package	Reserved Attempt to read/write will cause #UD.
1B0H	432	IA32_ENERGY_PERF_BIAS	Package	See Table 2-2.
F40H	3904	MSR_C8_PMON_BOX_CTRL	Package	Uncore C-box 8 perfmon local box control MSR.
F41H	3905	MSR_C8_PMON_BOX_STATUS	Package	Uncore C-box 8 perfmon local box status MSR.
F42H	3906	MSR_C8_PMON_BOX_OVF_CTRL	Package	Uncore C-box 8 perfmon local box overflow control MSR.
F50H	3920	MSR_C8_PMON_EVNT_SEL0	Package	Uncore C-box 8 perfmon event select MSR.
F51H	3921	MSR_C8_PMON_CTR0	Package	Uncore C-box 8 perfmon counter MSR.
F52H	3922	MSR_C8_PMON_EVNT_SEL1	Package	Uncore C-box 8 perfmon event select MSR.
F53H	3923	MSR_C8_PMON_CTR1	Package	Uncore C-box 8 perfmon counter MSR.
F54H	3924	MSR_C8_PMON_EVNT_SEL2	Package	Uncore C-box 8 perfmon event select MSR.
F55H	3925	MSR_C8_PMON_CTR2	Package	Uncore C-box 8 perfmon counter MSR.
F56H	3926	MSR_C8_PMON_EVNT_SEL3	Package	Uncore C-box 8 perfmon event select MSR.
F57H	3927	MSR_C8_PMON_CTR3	Package	Uncore C-box 8 perfmon counter MSR.
F58H	3928	MSR_C8_PMON_EVNT_SEL4	Package	Uncore C-box 8 perfmon event select MSR.
F59H	3929	MSR_C8_PMON_CTR4	Package	Uncore C-box 8 perfmon counter MSR.
F5AH	3930	MSR_C8_PMON_EVNT_SEL5	Package	Uncore C-box 8 perfmon event select MSR.
F5BH	3931	MSR_C8_PMON_CTR5	Package	Uncore C-box 8 perfmon counter MSR.
FCOH	4032	MSR_C9_PMON_BOX_CTRL	Package	Uncore C-box 9 perfmon local box control MSR.
FC1H	4033	MSR_C9_PMON_BOX_STATUS	Package	Uncore C-box 9 perfmon local box status MSR.
FC2H	4034	MSR_C9_PMON_BOX_OVF_CTRL	Package	Uncore C-box 9 perfmon local box overflow control MSR.
FD0H	4048	MSR_C9_PMON_EVNT_SEL0	Package	Uncore C-box 9 perfmon event select MSR.
FD1H	4049	MSR_C9_PMON_CTR0	Package	Uncore C-box 9 perfmon counter MSR.
FD2H	4050	MSR_C9_PMON_EVNT_SEL1	Package	Uncore C-box 9 perfmon event select MSR.
FD3H	4051	MSR_C9_PMON_CTR1	Package	Uncore C-box 9 perfmon counter MSR.

Table 2-19. Additional MSRs Supported by Intel® Xeon® Processor E7 Family (Contd.)

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
FD4H	4052	MSR_C9_PMON_EVNT_SEL2	Package	Uncore C-box 9 perfmon event select MSR.
FD5H	4053	MSR_C9_PMON_CTR2	Package	Uncore C-box 9 perfmon counter MSR.
FD6H	4054	MSR_C9_PMON_EVNT_SEL3	Package	Uncore C-box 9 perfmon event select MSR.
FD7H	4055	MSR_C9_PMON_CTR3	Package	Uncore C-box 9 perfmon counter MSR.
FD8H	4056	MSR_C9_PMON_EVNT_SEL4	Package	Uncore C-box 9 perfmon event select MSR.
FD9H	4057	MSR_C9_PMON_CTR4	Package	Uncore C-box 9 perfmon counter MSR.
FDAH	4058	MSR_C9_PMON_EVNT_SEL5	Package	Uncore C-box 9 perfmon event select MSR.
FDBH	4059	MSR_C9_PMON_CTR5	Package	Uncore C-box 9 perfmon counter MSR.

2.11 MSRS IN INTEL® PROCESSOR FAMILY BASED ON SANDY BRIDGE MICROARCHITECTURE

Table 2-20 lists model-specific registers (MSRs) that are common to the Intel $^{\textcircled{\$}}$ processor family based on Sandy Bridge microarchitecture. These processors have a CPUID signature with DisplayFamily_DisplayModel of 06_2AH, 06_2DH, see Table 2-1. Additional MSRs specific to 06_2AH are listed in Table 2-21.

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
OH	0	IA32_P5_MC_ADDR	Thread	See Section 2.23, "MSRs in Pentium Processors."
1H	1	IA32_P5_MC_TYPE	Thread	See Section 2.23, "MSRs in Pentium Processors."
6H	6	IA32_MONITOR_FILTER_ SIZE	Thread	See Section 9.10.5, "Monitor/Mwait Address Range Determination" and Table 2-2.
10H	16	IA32_TIME_STAMP_COUNTER	Thread	See Section 18.17, "Time-Stamp Counter" and see Table 2-2.
17H	23	IA32_PLATFORM_ID	Package	Platform ID (R) See Table 2-2.
1BH	27	IA32_APIC_BASE	Thread	See Section 11.4.4, "Local APIC Status and Location" and Table 2-2.
34H	52	MSR_SMI_COUNT	Thread	SMI Counter (R/O)
		31:0		SMI Count (R/O)
				Count SMIs.
		63:32		Reserved.

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
ЗАН	58	IA32_FEATURE_CONTROL	Thread	Control Features in Intel 64 Processor (R/W) See Table 2-2.
		0		Lock (R/WL)
		1		Enable VMX Inside SMX Operation (R/WL)
		2		Enable VMX Outside SMX Operation (R/WL)
		14:8		SENTER Local Functions Enables (R/WL)
		15		SENTER Global Functions Enable (R/WL)
79H	121	IA32_BIOS_UPDT_TRIG	Core	BIOS Update Trigger Register (W) See Table 2-2.
8BH	139	IA32_BIOS_SIGN_ID	Thread	BIOS Update Signature ID (R/W) See Table 2-2.
C1H	193	IA32_PMC0	Thread	Performance Counter Register See Table 2-2.
C2H	194	IA32_PMC1	Thread	Performance Counter Register See Table 2-2.
СЗН	195	IA32_PMC2	Thread	Performance Counter Register See Table 2-2.
C4H	196	IA32_PMC3	Thread	Performance Counter Register See Table 2-2.
C5H	197	IA32_PMC4	Core	Performance Counter Register (if core not shared by threads)
C6H	198	IA32_PMC5	Core	Performance Counter Register (if core not shared by threads)
C7H	199	IA32_PMC6	Core	Performance Counter Register (if core not shared by threads)
C8H	200	IA32_PMC7	Core	Performance Counter Register (if core not shared by threads)
CEH	206	MSR_PLATFORM_INFO	Package	Platform Information Contains power management and other model specific features enumeration. See http://biosbits.org.
		7:0		Reserved
		15:8	Package	Maximum Non-Turbo Ratio (R/O)
				This is the ratio of the frequency that invariant TSC runs at. Frequency = ratio * 100 MHz.
		27:16		Reserved
		28	Package	Programmable Ratio Limit for Turbo Mode (R/O)
				When set to 1, indicates that Programmable Ratio Limit for Turbo mode is enabled. When set to 0, indicates Programmable Ratio Limit for Turbo mode is disabled.

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		29	Package	Programmable TDP Limit for Turbo Mode (R/O) When set to 1, indicates that TDP Limit for Turbo mode is programmable. When set to 0, indicates TDP Limit for Turbo mode is not programmable.
		39:30		Reserved
		47:40	Package	Maximum Efficiency Ratio (R/O)
				This is the minimum ratio (maximum efficiency) that the processor can operate, in units of 100MHz.
		63:48		Reserved
E2H	226	MSR_PKG_CST_CONFIG_CONTROL	Соге	C-State Configuration Control (R/W)
				Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
				See http://biosbits.org.
		2:0		Package C-State Limit (R/W)
				Specifies the lowest processor-specific C-state code name (consuming the least power) for the package. The default is set as factory-configured package C-state limit.
				The following C-state code name encodings are supported:
				000b: CO/C1 (no package C-sate support)
				001b: C2 010b: C6 no retention
				011b: C6 retention
				100b: C7
				101b: C7s
				111: No package C-state limit
				Note: This field cannot be used to limit package C-state to C3.
		9:3		Reserved
		10		I/O MWAIT Redirection Enable (R/W)
				When set, will map IO_read instructions sent to IO register specified by MSR_PMG_IO_CAPTURE_BASE to MWAIT instructions.
		14:11		Reserved
		15		CFG Lock (R/W0)
				When set, locks bits 15:0 of this register until next reset.
		24:16		Reserved
		25		C3 State Auto Demotion Enable (R/W)
				When set, the processor will conditionally demote C6/C7 requests to C3 based on uncore auto-demote information.

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

Regi		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	7		
		26		C1 State Auto Demotion Enable (R/W)
				When set, the processor will conditionally demote C3/C6/C7 requests to C1 based on uncore auto-demote information.
		27		Enable C3 Undemotion (R/W)
				When set, enables undemotion from demoted C3.
		28		Enable C1 Undemotion (R/W)
				When set, enables undemotion from demoted C1.
		63:29		Reserved
E4H	228	MSR_PMG_IO_CAPTURE_BASE	Core	Power Management IO Redirection in C-state (R/W) See http://biosbits.org.
		15:0		LVL_2 Base Address (R/W)
				Specifies the base address visible to software for IO redirection. If IO MWAIT Redirection is enabled, reads to this address will be consumed by the power management logic and decoded to MWAIT instructions. When IO port address redirection is enabled, this is the IO port address reported to the OS/software.
		18:16		C-State Range (R/W)
				Specifies the encoding value of the maximum C-State code name to be included when IO read to MWAIT redirection is enabled by MSR_PKG_CST_CONFIG_CONTROL[bit10]:
				000b - C3 is the max C-State to include.
				001b - C6 is the max C-State to include.
				010b - C7 is the max C-State to include.
		63:19		Reserved
E7H	231	IA32_MPERF	Thread	Maximum Performance Frequency Clock Count (R/W) See Table 2-2.
E8H	232	IA32_APERF	Thread	Actual Performance Frequency Clock Count (R/W)
				See Table 2-2.
FEH	254	IA32_MTRRCAP	Thread	See Table 2-2.
13CH	316	MSR_FEATURE_CONFIG	Соге	AES Configuration (RW-L)
				Privileged post-BIOS agent must provide a #GP handler to handle unsuccessful read of this MSR.

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

Reg	ister Iress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		1:0		AES Configuration (RW-L)
				Upon a successful read of this MSR, the configuration of AES instruction set availability is as follows:
				11b: AES instructions are not available until next RESET.
				Otherwise, AES instructions are available.
				Note, AES instruction set is not available if read is unsuccessful. If the configuration is not 01b, AES instructions can be mis-configured if a privileged agent unintentionally writes 11b.
		63:2		Reserved
174H	372	IA32_SYSENTER_CS	Thread	See Table 2-2.
175H	373	IA32_SYSENTER_ESP	Thread	See Table 2-2.
176H	374	IA32_SYSENTER_EIP	Thread	See Table 2-2.
179H	377	IA32_MCG_CAP	Thread	See Table 2-2.
17AH	378	IA32_MCG_STATUS	Thread	Global Machine Check Status
				When set, bit indicates that the instruction addressed by the instruction pointer pushed on the stack (when the machine check was generated) can be used to restart the program. If cleared, the program cannot be reliably restarted.
		1		EIPV When set, bit indicates that the instruction addressed by the instruction pointer pushed on the stack (when the machine check was generated) is directly
				associated with the error.
		2		MCIP When set, bit indicates that a machine check has been generated. If a second machine check is detected while this bit is still set, the processor enters a shutdown state. Software should write this bit to 0 after processing a machine check exception.
		63:3		Reserved
186H	390	IA32_PERFEVTSEL0	Thread	See Table 2-2.
187H	391	IA32_PERFEVTSEL1	Thread	See Table 2-2.
188H	392	IA32_PERFEVTSEL2	Thread	See Table 2-2.
189H	393	IA32_PERFEVTSEL3	Thread	See Table 2-2.
18AH	394	IA32_PERFEVTSEL4	Core	See Table 2-2. If CPUID.0AH:EAX[15:8] > 4.
18BH	395	IA32_PERFEVTSEL5	Core	See Table 2-2. If CPUID.OAH:EAX[15:8] > 5.
18CH	396	IA32_PERFEVTSEL6	Core	See Table 2-2. If CPUID.OAH:EAX[15:8] > 6.
18DH	397	IA32_PERFEVTSEL7	Core	See Table 2-2. If CPUID.OAH:EAX[15:8] > 7.
198H	408	IA32_PERF_STATUS	Package	See Table 2-2.

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		15:0		Current Performance State Value
		63:16		Reserved
198H	408	MSR_PERF_STATUS	Package	Performance Status
		47:32		Core Voltage (R/O)
				P-state core voltage can be computed by
				MSR_PERF_STATUS[37:32] * (float) 1/(2^13).
199H	409	IA32_PERF_CTL	Thread	See Table 2-2.
19AH	410	IA32_CLOCK_MODULATION	Thread	Clock Modulation (R/W)
				See Table 2-2.
				IA32_CLOCK_MODULATION MSR was originally named IA32_THERM_CONTROL MSR.
		3:0		On demand Clock Modulation Duty Cycle (R/W)
				In 6.25% increment.
		4		On demand Clock Modulation Enable (R/W)
		63:5		Reserved
19BH	411	IA32_THERM_INTERRUPT	Core	Thermal Interrupt Control (R/W)
				See Table 2-2.
19CH	412	IA32_THERM_STATUS	Core	Thermal Monitor Status (R/W)
				See Table 2-2.
		0		Thermal Status (R/O)
				See Table 2-2.
		1		Thermal Status Log (R/WC0)
				See Table 2-2.
		2		PROTCHOT # or FORCEPR# Status (R/O)
				See Table 2-2.
		3		PROTCHOT # or FORCEPR# Log (R/WCO)
				See Table 2-2.
		4		Critical Temperature Status (R/O) See Table 2-2.
		5		
		٥		Critical Temperature Status Log (R/WC0) See Table 2-2.
		6		Thermal Threshold #1 Status (R/O)
				See Table 2-2.
		7		Thermal Threshold #1 Log (R/WC0)
		ĺ		See Table 2-2.
		8		Thermal Threshold #2 Status (R/O)
				See Table 2-2.
		9		Thermal Threshold #2 Log (R/WC0)
				See Table 2-2.

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		10		Power Limitation Status (R/O) See Table 2-2.
		11		
				Power Limitation Log (R/WC0) See Table 2-2.
		15:12		Reserved
		22:16		Digital Readout (R/O) See Table 2-2.
		26:23		Reserved
		30:27		Resolution in Degrees Celsius (R/O)
				See Table 2-2.
		31		Reading Valid (R/0)
				See Table 2-2.
		63:32		Reserved
1A0H	416	IA32_MISC_ENABLE		Enable Misc. Processor Features (R/W)
				Allows a variety of processor functions to be enabled and disabled.
		0	Thread	Fast-Strings Enable
				See Table 2-2
		6:1		Reserved
		7	Thread	Performance Monitoring Available (R)
				See Table 2-2.
		10:8		Reserved
		11	Thread	Branch Trace Storage Unavailable (R/O) See Table 2-2.
		12	Thread	Processor Event Based Sampling Unavailable (R/O) See Table 2-2.
		15:13		Reserved
		16	Package	Enhanced Intel SpeedStep Technology Enable (R/W)
				See Table 2-2.
		18	Thread	ENABLE MONITOR FSM (R/W)
				See Table 2-2.
		21:19		Reserved
		22	Thread	Limit CPUID Maxval (R/W)
				See Table 2-2.
		23	Thread	xTPR Message Disable (R/W) See Table 2-2.
		33:24		Reserved
		34	Thread	XD Bit Disable (R/W)
			1111000	See Table 2-2.

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

Regi Addı		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	7		
		37:35		Reserved
		38	Package	Turbo Mode Disable (R/W)
				When set to 1 on processors that support Intel Turbo Boost Technology, the turbo mode feature is disabled and the IDA_Enable feature flag will be clear (CPUID.06H: EAX[1]=0).
				When set to a 0 on processors that support IDA, CPUID.06H: EAX[1] reports the processor's support of turbo mode is enabled.
				Note: The power-on default value is used by BIOS to detect hardware support of turbo mode. If the power-on default value is 1, turbo mode is available in the processor. If the power-on default value is 0, turbo mode is not available.
		63:39		Reserved
1A2H	418	MSR_TEMPERATURE_TARGET	Unique	Temperature Target
		15:0		Reserved
		23:16		Temperature Target (R)
				The minimum temperature at which PROCHOT# will be asserted. The value is degrees C.
		63:24		Reserved
1A4H	420	MSR_MISC_FEATURE_CONTROL		Miscellaneous Feature Control (R/W)
		0	Core	L2 Hardware Prefetcher Disable (R/W)
				If 1, disables the L2 hardware prefetcher, which fetches additional lines of code or data into the L2 cache.
		1	Core	L2 Adjacent Cache Line Prefetcher Disable (R/W)
				If 1, disables the adjacent cache line prefetcher, which fetches the cache line that comprises a cache line pair (128 bytes).
		2	Core	DCU Hardware Prefetcher Disable (R/W)
				If 1, disables the L1 data cache prefetcher, which fetches the next cache line into L1 data cache.
		3	Core	DCU IP Prefetcher Disable (R/W)
				If 1, disables the L1 data cache IP prefetcher, which uses sequential load history (based on instruction pointer of previous loads) to determine whether to prefetch additional lines.
		63:4		Reserved
1A6H	422	MSR_OFFCORE_RSP_0	Thread	Offcore Response Event Select Register (R/W)
1A7H	422	MSR_OFFCORE_RSP_1	Thread	Offcore Response Event Select Register (R/W)
1AAH	426	MSR_MISC_PWR_MGMT		Miscellaneous Power Management Control
				Various model specific features enumeration. See http://biosbits.org.

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

Regi Add	ster	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
1B0H	432	IA32_ENERGY_PERF_BIAS	Package	See Table 2-2.
1B1H	433	IA32_PACKAGE_THERM_STATUS	Package	See Table 2-2.
1B2H	434	IA32_PACKAGE_THERM_INTERRUPT	Package	See Table 2-2.
1C8H	456	MSR_LBR_SELECT	Thread	Last Branch Record Filtering Select Register (R/W)
				See Section 18.9.2, "Filtering of Last Branch Records."
		0		CPL_EQ_0
		1		CPL_NEQ_0
		2		JCC
		3		NEAR_REL_CALL
		4		NEAR_IND_CALL
		5		NEAR_RET
		6		NEAR_IND_JMP
		7		NEAR_REL_JMP
		8		FAR_BRANCH
		63:9		Reserved
1C9H	457	MSR_LASTBRANCH_TOS	Thread	Last Branch Record Stack TOS (R/W)
				Contains an index (bits 0-3) that points to the MSR containing the most recent branch record.
				See MSR_LASTBRANCH_0_FROM_IP (at 680H).
1D9H	473	IA32_DEBUGCTL	Thread	Debug Control (R/W)
				See Table 2-2.
		0		LBR: Last Branch Record
		1		BTF
		5:2		Reserved
		6		TR: Branch Trace
		7		BTS: Log Branch Trace Message to BTS buffer
		8		BTINT
		9		BTS_OFF_OS
		10		BTS_OFF_USER
		11		FREEZE_LBR_ON_PMI
		12		FREEZE_PERFMON_ON_PMI
		13		ENABLE_UNCORE_PMI
		14		FREEZE_WHILE_SMM
		63:15		Reserved
1DDH	477	MSR_LER_FROM_LIP	Thread	Last Exception Record From Linear IP (R/W)
				Contains a pointer to the last branch instruction that the processor executed prior to the last exception that was generated or the last interrupt that was handled.

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope Scope	Bit Description
Hex	Dec			
1DEH	478	MSR_LER_TO_LIP	Thread	Last Exception Record To Linear IP (R/W) This area contains a pointer to the target of the last branch instruction that the processor executed prior to the last exception that was generated or the last interrupt that was handled.
1F2H	498	IA32_SMRR_PHYSBASE	Core	See Table 2-2.
1F3H	499	IA32_SMRR_PHYSMASK	Core	See Table 2-2.
1FCH	508	MSR_POWER_CTL	Соге	See http://biosbits.org.
200H	512	IA32_MTRR_PHYSBASE0	Thread	See Table 2-2.
201H	513	IA32_MTRR_PHYSMASK0	Thread	See Table 2-2.
202H	514	IA32_MTRR_PHYSBASE1	Thread	See Table 2-2.
203H	515	IA32_MTRR_PHYSMASK1	Thread	See Table 2-2.
204H	516	IA32_MTRR_PHYSBASE2	Thread	See Table 2-2.
205H	517	IA32_MTRR_PHYSMASK2	Thread	See Table 2-2.
206H	518	IA32_MTRR_PHYSBASE3	Thread	See Table 2-2.
207H	519	IA32_MTRR_PHYSMASK3	Thread	See Table 2-2.
208H	520	IA32_MTRR_PHYSBASE4	Thread	See Table 2-2.
209H	521	IA32_MTRR_PHYSMASK4	Thread	See Table 2-2.
20AH	522	IA32_MTRR_PHYSBASE5	Thread	See Table 2-2.
20BH	523	IA32_MTRR_PHYSMASK5	Thread	See Table 2-2.
20CH	524	IA32_MTRR_PHYSBASE6	Thread	See Table 2-2.
20DH	525	IA32_MTRR_PHYSMASK6	Thread	See Table 2-2.
20EH	526	IA32_MTRR_PHYSBASE7	Thread	See Table 2-2.
20FH	527	IA32_MTRR_PHYSMASK7	Thread	See Table 2-2.
210H	528	IA32_MTRR_PHYSBASE8	Thread	See Table 2-2.
211H	529	IA32_MTRR_PHYSMASK8	Thread	See Table 2-2.
212H	530	IA32_MTRR_PHYSBASE9	Thread	See Table 2-2.
213H	531	IA32_MTRR_PHYSMASK9	Thread	See Table 2-2.
250H	592	IA32_MTRR_FIX64K_00000	Thread	See Table 2-2.
258H	600	IA32_MTRR_FIX16K_80000	Thread	See Table 2-2.
259H	601	IA32_MTRR_FIX16K_A0000	Thread	See Table 2-2.
268H	616	IA32_MTRR_FIX4K_C0000	Thread	See Table 2-2.
269H	617	IA32_MTRR_FIX4K_C8000	Thread	See Table 2-2.
26AH	618	IA32_MTRR_FIX4K_D0000	Thread	See Table 2-2.
26BH	619	IA32_MTRR_FIX4K_D8000	Thread	See Table 2-2.
26CH	620	IA32_MTRR_FIX4K_E0000	Thread	See Table 2-2.

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
26DH	621	IA32_MTRR_FIX4K_E8000	Thread	See Table 2-2.
26EH	622	IA32_MTRR_FIX4K_F0000	Thread	See Table 2-2.
26FH	623	IA32_MTRR_FIX4K_F8000	Thread	See Table 2-2.
277H	631	IA32_PAT	Thread	See Table 2-2.
280H	640	IA32_MC0_CTL2	Core	See Table 2-2.
281H	641	IA32_MC1_CTL2	Соге	See Table 2-2.
282H	642	IA32_MC2_CTL2	Соге	See Table 2-2.
283H	643	IA32_MC3_CTL2	Соге	See Table 2-2.
284H	644	IA32_MC4_CTL2	Package	Always 0 (CMCI not supported).
2FFH	767	IA32_MTRR_DEF_TYPE	Thread	Default Memory Types (R/W)
				See Table 2-2.
309H	777	IA32_FIXED_CTR0	Thread	Fixed-Function Performance Counter Register 0 (R/W)
				See Table 2-2.
30AH	778	IA32_FIXED_CTR1	Thread	Fixed-Function Performance Counter Register 1 (R/W)
				See Table 2-2.
30BH	779	IA32_FIXED_CTR2	Thread	Fixed-Function Performance Counter Register 2 (R/W)
			<u> </u>	See Table 2-2.
345H	837	IA32_PERF_CAPABILITIES	Thread	See Table 2-2. See Section 18.4.1, "IA32_DEBUGCTL MSR."
		5:0		LBR Format
				See Table 2-2.
		6		PEBS Record Format.
		7		PEBSSaveArchRegs
				See Table 2-2.
		11:8		PEBS_REC_FORMAT
				See Table 2-2.
		12		SMM_FREEZE
				See Table 2-2.
		63:13		Reserved
38DH	909	IA32_FIXED_CTR_CTRL	Thread	Fixed-Function-Counter Control Register (R/W) See Table 2-2.
38EH	910	IA32_PERF_GLOBAL_STATUS		See Table 2-2. See Section 20.6.2.2, "Global Counter Control Facilities."
		0	Thread	Ovf_PMCO
		1	Thread	Ovf_PMC1
		2	Thread	Ovf_PMC2
		3	Thread	Ovf_PMC3
		4	Core	Ovf_PMC4 (if CPUID.0AH:EAX[15:8] > 4)

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

Regi Add	ister	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		5	Соге	Ovf_PMC5 (if CPUID.0AH:EAX[15:8] > 5)
		6	Соге	Ovf_PMC6 (if CPUID.OAH:EAX[15:8] > 6)
		7	Core	Ovf_PMC7 (if CPUID.OAH:EAX[15:8] > 7)
		31:8		Reserved
		32	Thread	Ovf_FixedCtr0
		33	Thread	Ovf_FixedCtr1
		34	Thread	Ovf_FixedCtr2
		60:35		Reserved
		61	Thread	Ovf_Uncore
		62	Thread	Ovf_BufDSSAVE
		63	Thread	CondChgd
38FH	911	IA32_PERF_GLOBAL_CTRL	Thread	See Table 2-2. See Section 20.6.2.2, "Global Counter Control Facilities."
		0	Thread	Set 1 to enable PMC0 to count.
		1	Thread	Set 1 to enable PMC1 to count.
		2	Thread	Set 1 to enable PMC2 to count.
		3	Thread	Set 1 to enable PMC3 to count.
		4	Core	Set 1 to enable PMC4 to count (if CPUID.0AH:EAX[15:8] > 4).
		5	Core	Set 1 to enable PMC5 to count (if CPUID.0AH:EAX[15:8] > 5).
		6	Core	Set 1 to enable PMC6 to count (if CPUID.0AH:EAX[15:8] > 6).
		7	Core	Set 1 to enable PMC7 to count (if CPUID.0AH:EAX[15:8] > 7).
		31:8		Reserved
		32	Thread	Set 1 to enable FixedCtr0 to count.
		33	Thread	Set 1 to enable FixedCtr1 to count.
		34	Thread	Set 1 to enable FixedCtr2 to count.
		63:35		Reserved
390H	912	IA32_PERF_GLOBAL_OVF_CTRL		See Table 2-2. See Section 20.6.2.2, "Global Counter Control Facilities."
		0	Thread	Set 1 to clear Ovf_PMCO.
		1	Thread	Set 1 to clear Ovf_PMC1.
		2	Thread	Set 1 to clear Ovf_PMC2.
		3	Thread	Set 1 to clear Ovf_PMC3.
		4	Соге	Set 1 to clear Ovf_PMC4 (if CPUID.0AH:EAX[15:8] > 4).
		5	Core	Set 1 to clear Ovf_PMC5 (if CPUID.0AH:EAX[15:8] > 5).
		6	Соге	Set 1 to clear Ovf_PMC6 (if CPUID.0AH:EAX[15:8] > 6).

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		7	Core	Set 1 to clear Ovf_PMC7 (if CPUID.0AH:EAX[15:8] > 7).
		31:8		Reserved
		32	Thread	Set 1 to clear Ovf_FixedCtr0.
		33	Thread	Set 1 to clear Ovf_FixedCtr1.
		34	Thread	Set 1 to clear Ovf_FixedCtr2.
		60:35		Reserved
		61	Thread	Set 1 to clear Ovf_Uncore.
		62	Thread	Set 1 to clear Ovf_BufDSSAVE.
		63	Thread	Set 1 to clear CondChgd.
3F1H	1009	IA32_PEBS_ENABLE (MSR_PEBS_ENABLE)	Thread	See Section 20.3.1.1.1, "Processor Event Based Sampling (PEBS)."
		0		Enable PEBS on IA32_PMCO. (R/W)
		1		Enable PEBS on IA32_PMC1. (R/W)
		2		Enable PEBS on IA32_PMC2. (R/W)
		3		Enable PEBS on IA32_PMC3. (R/W)
		31:4		Reserved
		32		Enable Load Latency on IA32_PMCO. (R/W)
		33		Enable Load Latency on IA32_PMC1. (R/W)
		34		Enable Load Latency on IA32_PMC2. (R/W)
		35		Enable Load Latency on IA32_PMC3. (R/W)
		62:36		Reserved
		63		Enable Precise Store (R/W)
3F6H	1014	MSR_PEBS_LD_LAT	Thread	See Section 20.3.1.1.2, "Load Latency Performance Monitoring Facility."
		15:0		Minimum threshold latency value of tagged load operation that will be counted. (R/W)
		63:36		Reserved
3F8H	1016	MSR_PKG_C3_RESIDENCY	Package	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		63:0		Package C3 Residency Counter (R/O)
				Value since last reset that this package is in processor-specific C3 states. Count at the same frequency as the TSC.
3F9H	1017	MSR_PKG_C6_RESIDENCY	Package	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		63:0		Package C6 Residency Counter. (R/O) Value since last reset that this package is in processor- specific C6 states. Count at the same frequency as the TSC.
3FAH	1018	MSR_PKG_C7_RESIDENCY	Package	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		63:0		Package C7 Residency Counter (R/O) Value since last reset that this package is in processor- specific C7 states. Count at the same frequency as the TSC.
3FCH	1020	MSR_CORE_C3_RESIDENCY	Core	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		63:0		CORE C3 Residency Counter (R/O)
				Value since last reset that this core is in processor- specific C3 states. Count at the same frequency as the TSC.
3FDH	1021	MSR_CORE_C6_RESIDENCY	Core	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		63:0		CORE C6 Residency Counter (R/O)
				Value since last reset that this core is in processor- specific C6 states. Count at the same frequency as the TSC.
3FEH	1022	MSR_CORE_C7_RESIDENCY	Core	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		63:0		CORE C7 Residency Counter (R/O)
				Value since last reset that this core is in processor- specific C7 states. Count at the same frequency as the TSC.
400H	1024	IA32_MCO_CTL	Соге	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
401H	1025	IA32_MCO_STATUS	Core	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
402H	1026	IA32_MCO_ADDR	Соге	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
403H	1027	IA32_MCO_MISC	Core	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
404H	1028	IA32_MC1_CTL	Core	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
405H	1029	IA32_MC1_STATUS	Core	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
406H	1030	IA32_MC1_ADDR	Соге	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
407H	1031	IA32_MC1_MISC	Core	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
408H	1032	IA32_MC2_CTL	Соге	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
409H	1033	IA32_MC2_STATUS	Core	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
40AH	1034	IA32_MC2_ADDR	Соге	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
40BH	1035	IA32_MC2_MISC	Соге	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
40CH	1036	IA32_MC3_CTL	Соге	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
40DH	1037	IA32_MC3_STATUS	Core	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
40EH	1038	IA32_MC3_ADDR	Core	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
40FH	1039	IA32_MC3_MISC	Соге	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
410H	1040	IA32_MC4_CTL	Соге	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
		0		PCU Hardware Error (R/W)
				When set, enables signaling of PCU hardware detected errors.
		1		PCU Controller Error (R/W)
				When set, enables signaling of PCU controller detected errors.
		2		PCU Firmware Error (R/W)
				When set, enables signaling of PCU firmware detected errors.
		63:2		Reserved
411H	1041	IA32_MC4_STATUS	Core	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
480H	1152	IA32_VMX_BASIC	Thread	Reporting Register of Basic VMX Capabilities (R/O)
				See Table 2-2.
				See Appendix A.1, "Basic VMX Information."
481H	1153	IA32_VMX_PINBASED_CTLS	Thread	Capability Reporting Register of Pin-Based VM-Execution Controls (R/O)
				See Table 2-2.
				See Appendix A.3, "VM-Execution Controls."
482H	1154	IA32_VMX_PROCBASED_CTLS	Thread	Capability Reporting Register of Primary Processor- Based VM-Execution Controls (R/O)
				See Appendix A.3, "VM-Execution Controls."
483H	1155	IA32_VMX_EXIT_CTLS	Thread	Capability Reporting Register of VM-Exit Controls (R/O)
				See Table 2-2.
				See Appendix A.4, "VM-Exit Controls."
484H	1156	IA32_VMX_ENTRY_CTLS	Thread	Capability Reporting Register of VM-Entry Controls (R/O)
				See Table 2-2.
				See Appendix A.5, "VM-Entry Controls."

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
485H	1157	IA32_VMX_MISC	Thread	Reporting Register of Miscellaneous VMX Capabilities (R/O) See Table 2-2.
				See Appendix A.6, "Miscellaneous Data."
486H	1158	IA32_VMX_CR0_FIXED0	Thread	Capability Reporting Register of CRO Bits Fixed to 0 (R/O) See Table 2-2. See Appendix A.7, "VMX-Fixed Bits in CRO."
487H	1159	IA32_VMX_CR0_FIXED1	Thread	Capability Reporting Register of CRO Bits Fixed to 1 (R/O) See Table 2-2. See Appendix A.7, "VMX-Fixed Bits in CRO."
488H	1160	IA32_VMX_CR4_FIXED0	Thread	Capability Reporting Register of CR4 Bits Fixed to 0 (R/0) See Table 2-2. See Appendix A.8, "VMX-Fixed Bits in CR4."
489H	1161	IA32_VMX_CR4_FIXED1	Thread	Capability Reporting Register of CR4 Bits Fixed to 1 (R/O) See Table 2-2. See Appendix A.8, "VMX-Fixed Bits in CR4."
48AH	1162	IA32_VMX_VMCS_ENUM	Thread	Capability Reporting Register of VMCS Field Enumeration (R/O) See Table 2-2. See Appendix A.9, "VMCS Enumeration."
48BH	1163	IA32_VMX_PROCBASED_CTLS2	Thread	Capability Reporting Register of Secondary Processor- Based VM-Execution Controls (R/O) See Appendix A.3, "VM-Execution Controls."
48CH	1164	IA32_VMX_EPT_VPID_ENUM	Thread	Capability Reporting Register of EPT and VPID (R/O) See Table 2-2
48DH	1165	IA32_VMX_TRUE_PINBASED_CTLS	Thread	Capability Reporting Register of Pin-Based VM-Execution Flex Controls (R/O) See Table 2-2
48EH	1166	IA32_VMX_TRUE_PROCBASED_CTLS	Thread	Capability Reporting Register of Primary Processor- Based VM-Execution Flex Controls (R/O) See Table 2-2
48FH	1167	IA32_VMX_TRUE_EXIT_CTLS	Thread	Capability Reporting Register of VM-Exit Flex Controls (R/O) See Table 2-2
490H	1168	IA32_VMX_TRUE_ENTRY_CTLS	Thread	Capability Reporting Register of VM-Entry Flex Controls (R/O) See Table 2-2
4C1H	1217	IA32_A_PMCO	Thread	See Table 2-2.
-	•		•	

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
4C2H	1218	IA32_A_PMC1	Thread	See Table 2-2.
4C3H	1219	IA32_A_PMC2	Thread	See Table 2-2.
4C4H	1220	IA32_A_PMC3	Thread	See Table 2-2.
4C5H	1221	IA32_A_PMC4	Соге	See Table 2-2.
4C6H	1222	IA32_A_PMC5	Соге	See Table 2-2.
4C7H	1223	IA32_A_PMC6	Core	See Table 2-2.
4C8H	1224	IA32_A_PMC7	Core	See Table 2-2.
600H	1536	IA32_DS_AREA	Thread	DS Save Area (R/W) See Table 2-2. See Section 20.6.3.4, "Debug Store (DS) Mechanism."
606H	1542	MSR_RAPL_POWER_UNIT	Package	Unit Multipliers used in RAPL Interfaces (R/0) See Section 15.10.1, "RAPL Interfaces."
60AH	1546	MSR_PKGC3_IRTL	Package	Package C3 Interrupt Response Limit (R/W) Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		9:0		Interrupt Response Time Limit (R/W) Specifies the limit that should be used to decide if the package should be put into a package C3 state.
		12:10		Time Unit (R/W) Specifies the encoding value of time unit of the interrupt response time limit. The following time unit encodings are supported: 000b: 1 ns 001b: 32 ns 010b: 1024 ns 011b: 32768 ns 100b: 1048576 ns
		1442		101b: 33554432 ns
		14:13		Reserved
		15		Valid (R/W) Indicates whether the values in bits 12:0 are valid and can be used by the processor for package C-sate management.
		63:16		Reserved

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
60BH	1547	MSR_PKGC6_IRTL	Package	Package C6 Interrupt Response Limit (R/W) This MSR defines the budget allocated for the package to exit from a C6 to a C0 state, where an interrupt request can be delivered to the core and serviced. Additional core-exit latency may be applicable depending on the actual C-state the core is in. Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state
				parameters or ACPI C-states.
		9:0		Interrupt Response Time Limit (R/W)
				Specifies the limit that should be used to decide if the package should be put into a package C6 state.
		12:10		Time Unit (R/W)
				Specifies the encoding value of time unit of the interrupt response time limit. The following time unit encodings are supported:
				000b: 1 ns
				001b: 32 ns
				010b: 1024 ns
				011b: 32768 ns
				100b: 1048576 ns
				101b: 33554432 ns
		14:13		Reserved
		15		Valid (R/W) Indicates whether the values in bits 12:0 are valid and can be used by the processor for package C-sate management.
		63:16		Reserved
60DH	1549	MSR_PKG_C2_RESIDENCY	Package	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		63:0		Package C2 Residency Counter (R/O) Value since last reset that this package is in processor- specific C2 states. Count at the same frequency as the TSC.
610H	1552	MSR_PKG_POWER_LIMIT	Package	PKG RAPL Power Limit Control (R/W)
				See Section 15.10.3, "Package RAPL Domain."
611H	1553	MSR_PKG_ENERGY_STATUS	Package	PKG Energy Status (R/O)
				See Section 15.10.3, "Package RAPL Domain."
614H	1556	MSR_PKG_POWER_INFO	Package	PKG RAPL Parameters (R/W) See Section 15.10.3, "Package RAPL Domain."
638H	1592	MSR_PP0_POWER_LIMIT	Package	PPO RAPL Power Limit Control (R/W) See Section 15.10.4, "PPO/PP1 RAPL Domains."

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
680H	1664	MSR_LASTBRANCH_0_FROM_IP	Thread	Last Branch Record O From IP (R/W)
				One of sixteen pairs of last branch record registers on the last branch record stack. This part of the stack contains pointers to the source instruction. See also:
				Last Branch Record Stack TOS at 1C9H.Section 18.9.1 and record format in Section 18.4.8.1.
681H	1665	MSR_LASTBRANCH_1_FROM_IP	Thread	Last Branch Record 1 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
682H	1666	MSR_LASTBRANCH_2_FROM_IP	Thread	Last Branch Record 2 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
683H	1667	MSR_LASTBRANCH_3_FROM_IP	Thread	Last Branch Record 3 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
684H	1668	MSR_LASTBRANCH_4_FROM_IP	Thread	Last Branch Record 4 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
685H	1669	MSR_LASTBRANCH_5_FROM_IP	Thread	Last Branch Record 5 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
686H	1670	MSR_LASTBRANCH_6_FROM_IP	Thread	Last Branch Record 6 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
687H	1671	MSR_LASTBRANCH_7_FROM_IP	Thread	Last Branch Record 7 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
688H	1672	MSR_LASTBRANCH_8_FROM_IP	Thread	Last Branch Record 8 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
689H	1673	MSR_LASTBRANCH_9_FROM_IP	Thread	Last Branch Record 9 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
68AH	1674	MSR_LASTBRANCH_10_FROM_IP	Thread	Last Branch Record 10 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
68BH	1675	MSR_LASTBRANCH_11_FROM_IP	Thread	Last Branch Record 11 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
68CH	1676	MSR_LASTBRANCH_12_FROM_IP	Thread	Last Branch Record 12 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
68DH	1677	MSR_LASTBRANCH_13_FROM_IP	Thread	Last Branch Record 13 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
68EH	1678	MSR_LASTBRANCH_14_FROM_IP	Thread	Last Branch Record 14 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
68FH	1679	MSR_LASTBRANCH_15_FROM_IP	Thread	Last Branch Record 15 From IP (R/W)
				See description of MSR_LASTBRANCH_0_FROM_IP.
6C0H	1728	MSR_LASTBRANCH_0_TO_IP	Thread	Last Branch Record O To IP (R/W)
				One of sixteen pairs of last branch record registers on the last branch record stack. This part of the stack contains pointers to the destination instruction.

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
6C1H	1729	MSR_LASTBRANCH_1_TO_IP	Thread	Last Branch Record 1 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6C2H	1730	MSR_LASTBRANCH_2_TO_IP	Thread	Last Branch Record 2 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6C3H	1731	MSR_LASTBRANCH_3_TO_IP	Thread	Last Branch Record 3 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6C4H	1732	MSR_LASTBRANCH_4_TO_IP	Thread	Last Branch Record 4 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6C5H	1733	MSR_LASTBRANCH_5_TO_IP	Thread	Last Branch Record 5 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6C6H	1734	MSR_LASTBRANCH_6_TO_IP	Thread	Last Branch Record 6 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6C7H	1735	MSR_LASTBRANCH_7_TO_IP	Thread	Last Branch Record 7 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6C8H	1736	MSR_LASTBRANCH_8_TO_IP	Thread	Last Branch Record 8 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6C9H	1737	MSR_LASTBRANCH_9_TO_IP	Thread	Last Branch Record 9 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6CAH	1738	MSR_LASTBRANCH_10_T0_IP	Thread	Last Branch Record 10 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6CBH	1739	MSR_LASTBRANCH_11_TO_IP	Thread	Last Branch Record 11 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6CCH	1740	MSR_LASTBRANCH_12_TO_IP	Thread	Last Branch Record 12 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6CDH	1741	MSR_LASTBRANCH_13_TO_IP	Thread	Last Branch Record 13 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6CEH	1742	MSR_LASTBRANCH_14_TO_IP	Thread	Last Branch Record 14 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6CFH	1743	MSR_LASTBRANCH_15_TO_IP	Thread	Last Branch Record 15 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6E0H	1760	IA32_TSC_DEADLINE	Thread	See Table 2-2.
802H- 83FH	2050- 2111	X2APIC MSRs	Thread	See Table 2-2.
0080H		IA32_EFER	Thread	Extended Feature Enables See Table 2-2.
C000_ 0081H		IA32_STAR	Thread	System Call Target Address (R/W) See Table 2-2.
C000_ 0082H		IA32_LSTAR	Thread	IA-32e Mode System Call Target Address (R/W) See Table 2-2.

Table 2-20. MSRs Supported by Intel® Processors Based on Sandy Bridge Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
C000_		IA32_FMASK	Thread	System Call Flag Mask (R/W)
0084H				See Table 2-2.
C000_		IA32_FS_BASE	Thread	Map of BASE Address of FS (R/W)
0100H				See Table 2-2.
C000_		IA32_GS_BASE	Thread	Map of BASE Address of GS (R/W)
0101H				See Table 2-2.
C000_		IA32_KERNEL_GS_BASE	Thread	Swap Target of BASE Address of GS (R/W)
0102H				See Table 2-2.
C000_		IA32_TSC_AUX	Thread	AUXILIARY TSC Signature (R/W)
0103H				See Table 2-2 and Section 18.17.2, "IA32_TSC_AUX Register and RDTSCP Support."

2.11.1 MSRs in the 2nd Generation Intel[®] Core[™] Processor Family Based on Sandy Bridge Microarchitecture

Table 2-21 and Table 2-22 list model-specific registers (MSRs) that are specific to the 2nd generation Intel[®] CoreTM processor family based on the Sandy Bridge microarchitecture. These processors have a CPUID signature with DisplayFamily_DisplayModel of 06_2AH ; see Table 2-1.

Table 2-21. MSRs Supported by the 2nd Generation Intel® Core™ Processors (Sandy Bridge Microarchitecture)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
1ADH	429	MSR_TURBO_RATIO_LIMIT	Package	Maximum Ratio Limit of Turbo Mode R/O if MSR_PLATFORM_INFO.[28] = 0. R/W if MSR_PLATFORM_INFO.[28] = 1.
		7:0	Package	Maximum Ratio Limit for 1C Maximum turbo ratio limit of 1 core active.
		15:8	Package	Maximum Ratio Limit for 2C Maximum turbo ratio limit of 2 core active.
		23:16	Package	Maximum Ratio Limit for 3C Maximum turbo ratio limit of 3 core active.
		31:24	Package	Maximum Ratio Limit for 4C Maximum turbo ratio limit of 4 core active.
		63:32		Reserved

Table 2-21. MSRs Supported by the 2nd Generation Intel® Core™ Processors (Sandy Bridge Microarchitecture)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
60CH	1548	MSR_PKGC7_IRTL	Package	Package C7 Interrupt Response Limit (R/W) This MSR defines the budget allocated for the package to exit from a C7 to a C0 state, where interrupt request can be delivered to the core and serviced. Additional core-exit latency may be applicable depending on the actual C-state the core is in. Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-states.
		9:0		Interrupt Response Time Limit (R/W) Specifies the limit that should be used to decide if the package should be put into a package C7 state.
		12:10		Time Unit (R/W) Specifies the encoding value of time unit of the interrupt response time limit. The following time unit encodings are supported: 000b: 1 ns 001b: 32 ns 010b: 1024 ns 011b: 32768 ns 100b: 1048576 ns 101b: 33554432 ns
		14:13		Reserved Valid (R/W) Indicates whether the values in bits 12:0 are valid and can be used by the processor for package C-sate management.
		63:16		Reserved
639H	1593	MSR_PPO_ENERGY_STATUS	Package	PP0 Energy Status (R/O) See Section 15.10.4, "PP0/PP1 RAPL Domains."
63AH	1594	MSR_PPO_POLICY	Package	PPO Balance Policy (R/W) See Section 15.10.4, "PPO/PP1 RAPL Domains."
640H	1600	MSR_PP1_POWER_LIMIT	Package	PP1 RAPL Power Limit Control (R/W) See Section 15.10.4, "PP0/PP1 RAPL Domains."
641H	1601	MSR_PP1_ENERGY_STATUS	Package	PP1 Energy Status (R/O) See Section 15.10.4, "PP0/PP1 RAPL Domains."
642H	1602	MSR_PP1_POLICY	Package	PP1 Balance Policy (R/W) See Section 15.10.4, "PP0/PP1 RAPL Domains."
See Table	e 2-20, Ta	able 2-21, and Table 2-22 for MSR definiti	ons applicable to p	processors with CPUID signature 06_2AH.

Table 2-22 lists the MSRs of uncore PMU for Intel processors with CPUID signature 06_2AH.

Table 2-22. Uncore PMU MSRs Supported by 2nd Generation Intel® Core™ Processors

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	1		
391H	913	MSR_UNC_PERF_GLOBAL_CTRL	Package	Uncore PMU Global Control
		0		Slice 0 select.
		1		Slice 1 select.
		2		Slice 2 select.
		3		Slice 3 select.
		4		Slice 4 select.
		18:5		Reserved
		29		Enable all uncore counters.
		30		Enable wake on PMI.
		31		Enable Freezing counter when overflow.
		63:32		Reserved
392H	914	MSR_UNC_PERF_GLOBAL_STATUS	Package	Uncore PMU Main Status
		0		Fixed counter overflowed.
		1		An ARB counter overflowed.
		2		Reserved
		3		A CBox counter overflowed (on any slice).
		63:4		Reserved
394H	916	MSR_UNC_PERF_FIXED_CTRL	Package	Uncore Fixed Counter Control (R/W)
		19:0		Reserved
		20		Enable overflow propagation.
		21		Reserved
		22		Enable counting.
		63:23		Reserved
395H	917	MSR_UNC_PERF_FIXED_CTR	Package	Uncore Fixed Counter
		47:0		Current count.
		63:48		Reserved
396H	918	MSR_UNC_CBO_CONFIG	Package	Uncore C-Box Configuration Information (R/O)
		3:0		Report the number of C-Box units with performance counters, including processor cores and processor graphics.
		63:4		Reserved
3B0H	946	MSR_UNC_ARB_PERFCTR0	Package	Uncore Arb Unit, Performance Counter 0
3B1H	947	MSR_UNC_ARB_PERFCTR1	Package	Uncore Arb Unit, Performance Counter 1
3B2H	944	MSR_UNC_ARB_PERFEVTSEL0	Package	Uncore Arb Unit, Counter 0 Event Select MSR
3B3H	945	MSR_UNC_ARB_PERFEVTSEL1	Package	Uncore Arb unit, Counter 1 Event Select MSR

Table 2-22. Uncore PMU MSRs Supported by 2nd Generation Intel® Core™ Processors

Regi Add	ster	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
700H	1792	MSR_UNC_CBO_0_PERFEVTSEL0	Package	Uncore C-Box O, Counter O Event Select MSR
701H	1793	MSR_UNC_CBO_0_PERFEVTSEL1	Package	Uncore C-Box O, Counter 1 Event Select MSR
702H	1794	MSR_UNC_CBO_0_PERFEVTSEL2	Package	Uncore C-Box O, Counter 2 Event Select MSR
703H	1795	MSR_UNC_CBO_0_PERFEVTSEL3	Package	Uncore C-Box O, Counter 3 Event Select MSR
705H	1797	MSR_UNC_CBO_O_UNIT_STATUS	Package	Uncore C-Box 0, Unit Status for Counter 0-3
706H	1798	MSR_UNC_CBO_0_PERFCTR0	Package	Uncore C-Box O, Performance Counter O
707H	1799	MSR_UNC_CBO_0_PERFCTR1	Package	Uncore C-Box O, Performance Counter 1
708H	1800	MSR_UNC_CBO_0_PERFCTR2	Package	Uncore C-Box O, Performance Counter 2
709H	1801	MSR_UNC_CBO_0_PERFCTR3	Package	Uncore C-Box O, Performance Counter 3
710H	1808	MSR_UNC_CBO_1_PERFEVTSELO	Package	Uncore C-Box 1, Counter 0 Event Select MSR
711H	1809	MSR_UNC_CBO_1_PERFEVTSEL1	Package	Uncore C-Box 1, Counter 1 Event Select MSR
712H	1810	MSR_UNC_CBO_1_PERFEVTSEL2	Package	Uncore C-Box 1, Counter 2 Event Select MSR
713H	1811	MSR_UNC_CBO_1_PERFEVTSEL3	Package	Uncore C-Box 1, Counter 3 Event Select MSR
715H	1813	MSR_UNC_CBO_1_UNIT_STATUS	Package	Uncore C-Box 1, Unit Status for Counter 0-3
716H	1814	MSR_UNC_CBO_1_PERFCTR0	Package	Uncore C-Box 1, Performance Counter 0
717H	1815	MSR_UNC_CBO_1_PERFCTR1	Package	Uncore C-Box 1, Performance Counter 1
718H	1816	MSR_UNC_CBO_1_PERFCTR2	Package	Uncore C-Box 1, Performance Counter 2
719H	1817	MSR_UNC_CBO_1_PERFCTR3	Package	Uncore C-Box 1, Performance Counter 3
720H	1824	MSR_UNC_CBO_2_PERFEVTSELO	Package	Uncore C-Box 2, Counter 0 Event Select MSR
721H	1825	MSR_UNC_CBO_2_PERFEVTSEL1	Package	Uncore C-Box 2, Counter 1 Event Select MSR
722H	1826	MSR_UNC_CBO_2_PERFEVTSEL2	Package	Uncore C-Box 2, Counter 2 Event Select MSR
723H	1827	MSR_UNC_CBO_2_PERFEVTSEL3	Package	Uncore C-Box 2, Counter 3 Event Select MSR
725H	1829	MSR_UNC_CBO_2_UNIT_STATUS	Package	Uncore C-Box 2, Unit Status for Counter 0-3
726H	1830	MSR_UNC_CBO_2_PERFCTRO	Package	Uncore C-Box 2, Performance Counter 0
727H	1831	MSR_UNC_CBO_2_PERFCTR1	Package	Uncore C-Box 2, Performance Counter 1
728H	1832	MSR_UNC_CBO_3_PERFCTR2	Package	Uncore C-Box 3, Performance Counter 2
729H	1833	MSR_UNC_CBO_3_PERFCTR3	Package	Uncore C-Box 3, Performance Counter 3
730H	1840	MSR_UNC_CBO_3_PERFEVTSELO	Package	Uncore C-Box 3, Counter 0 Event Select MSR
731H	1841	MSR_UNC_CBO_3_PERFEVTSEL1	Package	Uncore C-Box 3, Counter 1 Event Select MSR
732H	1842	MSR_UNC_CBO_3_PERFEVTSEL2	Package	Uncore C-Box 3, Counter 2 Event Select MSR
733H	1843	MSR_UNC_CBO_3_PERFEVTSEL3	Package	Uncore C-Box 3, counter 3 Event Select MSR
735H	1845	MSR_UNC_CBO_3_UNIT_STATUS	Package	Uncore C-Box 3, Unit Status for Counter 0-3
736H	1846	MSR_UNC_CBO_3_PERFCTRO	Package	Uncore C-Box 3, Performance Counter 0
737H	1847	MSR_UNC_CBO_3_PERFCTR1	Package	Uncore C-Box 3, Performance Counter 1
738H	1848	MSR_UNC_CBO_3_PERFCTR2	Package	Uncore C-Box 3, Performance Counter 2
739H	1849	MSR_UNC_CBO_3_PERFCTR3	Package	Uncore C-Box 3, Performance Counter 3
740H	1856	MSR_UNC_CBO_4_PERFEVTSEL0	Package	Uncore C-Box 4, Counter 0 Event Select MSR

Table 2-22. Uncore PMU MSRs Supported by 2nd Generation Intel® Core™ Processors

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
741H	1857	MSR_UNC_CBO_4_PERFEVTSEL1	Package	Uncore C-Box 4, Counter 1 Event Select MSR
742H	1858	MSR_UNC_CBO_4_PERFEVTSEL2	Package	Uncore C-Box 4, Counter 2 Event Select MSR
743H	1859	MSR_UNC_CBO_4_PERFEVTSEL3	Package	Uncore C-Box 4, Counter 3 Event Select MSR
745H	1861	MSR_UNC_CBO_4_UNIT_STATUS	Package	Uncore C-Box 4, Unit status for Counter 0-3
746H	1862	MSR_UNC_CBO_4_PERFCTRO	Package	Uncore C-Box 4, Performance Counter 0
747H	1863	MSR_UNC_CBO_4_PERFCTR1	Package	Uncore C-Box 4, Performance Counter 1
748H	1864	MSR_UNC_CBO_4_PERFCTR2	Package	Uncore C-Box 4, Performance Counter 2
749H	1865	MSR_UNC_CBO_4_PERFCTR3	Package	Uncore C-Box 4, Performance Counter 3

2.11.2 MSRs in the Intel® Xeon® Processor E5 Family Based on Sandy Bridge Microarchitecture

Table 2-23 lists additional model-specific registers (MSRs) that are specific to the Intel $^{(8)}$ Xeon $^{(8)}$ Processor E5 Family based on the Sandy Bridge microarchitecture. These processors have a CPUID signature with DisplayFamily_DisplayModel of 06_2DH, and also support MSRs listed in Table 2-20 and Table 2-24.

Table 2-23. Selected MSRs Supported by Intel® Xeon® Processors E5 Family Based on Sandy Bridge Microarchitecture

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
17FH	383	MSR_ERROR_CONTROL	Package	MC Bank Error Configuration (R/W)
		0		Reserved
		1		MemError Log Enable (R/W)
				When set, enables IMC status bank to log additional info in bits 36:32.
		63:2		Reserved
1ADH	429	MSR_TURBO_RATIO_LIMIT	Package	Maximum Ratio Limit of Turbo Mode
				R/O if MSR_PLATFORM_INFO.[28] = 0,
				R/W if MSR_PLATFORM_INFO.[28] = 1
		7:0	Package	Maximum Ratio Limit for 1C
				Maximum turbo ratio limit of 1 core active.
		15:8	Package	Maximum Ratio Limit for 2C
				Maximum turbo ratio limit of 2 cores active.
		23:16	Package	Maximum Ratio Limit for 3C
				Maximum turbo ratio limit of 3 cores active.
		31:24	Package	Maximum Ratio Limit for 4C
				Maximum turbo ratio limit of 4 cores active.
		39:32	Package	Maximum Ratio Limit for 5C
				Maximum turbo ratio limit of 5 cores active.

Table 2-23. Selected MSRs Supported by Intel® Xeon® Processors E5 Family Based on Sandy Bridge Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		47:40	Package	Maximum Ratio Limit for 6C
				Maximum turbo ratio limit of 6 cores active.
		55:48	Package	Maximum Ratio Limit for 7C
				Maximum turbo ratio limit of 7 cores active.
		63:56	Package	Maximum Ratio Limit for 8C
				Maximum turbo ratio limit of 8 cores active.
285H	645	IA32_MC5_CTL2	Package	See Table 2-2.
286H	646	IA32_MC6_CTL2	Package	See Table 2-2.
287H	647	IA32_MC7_CTL2	Package	See Table 2-2.
288H	648	IA32_MC8_CTL2	Package	See Table 2-2.
289H	649	IA32_MC9_CTL2	Package	See Table 2-2.
28AH	650	IA32_MC10_CTL2	Package	See Table 2-2.
28BH	651	IA32_MC11_CTL2	Package	See Table 2-2.
28CH	652	IA32_MC12_CTL2	Package	See Table 2-2.
28DH	653	IA32_MC13_CTL2	Package	See Table 2-2.
28EH	654	IA32_MC14_CTL2	Package	See Table 2-2.
28FH	655	IA32_MC15_CTL2	Package	See Table 2-2.
290H	656	IA32_MC16_CTL2	Package	See Table 2-2.
291H	657	IA32_MC17_CTL2	Package	See Table 2-2.
292H	658	IA32_MC18_CTL2	Package	See Table 2-2.
293H	659	IA32_MC19_CTL2	Package	See Table 2-2.
39CH	924	MSR_PEBS_NUM_ALT	Package	ENABLE_PEBS_NUM_ALT (R/W)
		0		ENABLE_PEBS_NUM_ALT (R/W)
				Write 1 to enable alternate PEBS counting logic for specific events requiring additional configuration, see https://perfmon-events.intel.com/ .
		63:1		Reserved, must be zero.
414H	1044	IA32_MC5_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
415H	1045	IA32_MC5_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
416H	1046	IA32_MC5_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
417H	1047	IA32_MC5_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
418H	1048	IA32_MC6_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
419H	1049	IA32_MC6_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
41AH	1050	IA32_MC6_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
41BH	1051	IA32_MC6_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
41CH	1052	IA32_MC7_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."

Table 2-23. Selected MSRs Supported by Intel® Xeon® Processors E5 Family Based on Sandy Bridge Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
41DH	1053	IA32_MC7_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
41EH	1054	IA32_MC7_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
41FH	1055	IA32_MC7_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
420H	1056	IA32_MC8_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
421H	1057	IA32_MC8_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
422H	1058	IA32_MC8_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
423H	1059	IA32_MC8_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
424H	1060	IA32_MC9_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
425H	1061	IA32_MC9_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
426H	1062	IA32_MC9_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
427H	1063	IA32_MC9_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
428H	1064	IA32_MC10_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
429H	1065	IA32_MC10_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
42AH	1066	IA32_MC10_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
42BH	1067	IA32_MC10_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
42CH	1068	IA32_MC11_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
42DH	1069	IA32_MC11_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
42EH	1070	IA32_MC11_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
42FH	1071	IA32_MC11_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
430H	1072	IA32_MC12_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
431H	1073	IA32_MC12_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
432H	1074	IA32_MC12_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
433H	1075	IA32_MC12_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
434H	1076	IA32_MC13_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
435H	1077	IA32_MC13_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
436H	1078	IA32_MC13_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
437H	1079	IA32_MC13_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
438H	1080	IA32_MC14_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
439H	1081	IA32_MC14_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
43AH	1082	IA32_MC14_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
43BH	1083	IA32_MC14_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."

Table 2-23. Selected MSRs Supported by Intel® Xeon® Processors E5 Family Based on Sandy Bridge Microarchitecture (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
43CH	1084	IA32_MC15_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
43DH	1085	IA32_MC15_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
43EH	1086	IA32_MC15_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
43FH	1087	IA32_MC15_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
440H	1088	IA32_MC16_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
441H	1089	IA32_MC16_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
442H	1090	IA32_MC16_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
443H	1091	IA32_MC16_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
444H	1092	IA32_MC17_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
445H	1093	IA32_MC17_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
446H	1094	IA32_MC17_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
447H	1095	IA32_MC17_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
448H	1096	IA32_MC18_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
449H	1097	IA32_MC18_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
44AH	1098	IA32_MC18_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
44BH	1099	IA32_MC18_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
44CH	1100	IA32_MC19_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
44DH	1101	IA32_MC19_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS" and Chapter 17.
44EH	1102	IA32_MC19_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
44FH	1103	IA32_MC19_MISC	Package	See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
613H	1555	MSR_PKG_PERF_STATUS	Package	Package RAPL Perf Status (R/O)
618H	1560	MSR_DRAM_POWER_LIMIT	Package	DRAM RAPL Power Limit Control (R/W)
				See Section 15.10.5, "DRAM RAPL Domain."
619H	1561	MSR_DRAM_ENERGY_STATUS	Package	DRAM Energy Status (R/O)
				See Section 15.10.5, "DRAM RAPL Domain."
61BH	1563	MSR_DRAM_PERF_STATUS	Package	DRAM Performance Throttling Status (R/O)
64.61.1	4504	MSD DD 444 DOL (5D 19150		See Section 15.10.5, "DRAM RAPL Domain."
61CH	1564	MSR_DRAM_POWER_INFO	Package	DRAM RAPL Parameters (R/W) See Section 15.10.5, "DRAM RAPL Domain."
639H	1593	MSR_PPO_ENERGY_STATUS	Package	PPO Energy Status (R/O)
חבכט	1722	I DIV_FFU_CINCNUT_31M1U3	rackaye	See Section 15.10.4, "PPO/PP1 RAPL Domains."
See Table	L e 2-20, Ta	ı able 2-23, and Table 2-24 for MSR defini	tions applicable to	processors with CPUID signature 06_2DH.

2.11.3 Additional Uncore PMU MSRs in the Intel® Xeon® Processor E5 Family

Intel Xeon Processor E5 family is based on the Sandy Bridge microarchitecture. The MSR-based uncore PMU interfaces are listed in Table 2-24. For complete detail of the uncore PMU, refer to Intel Xeon Processor E5 Product Family Uncore Performance Monitoring Guide. These processors have a CPUID signature with DisplayFamily_DisplayModel of 06_2DH

Table 2-24. Uncore PMU MSRs in Intel® Xeon® Processor E5 Family

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
C08H	3080	MSR_U_PMON_UCLK_FIXED_CTL	Package	Uncore U-box UCLK Fixed Counter Control
CO9H	3081	MSR_U_PMON_UCLK_FIXED_CTR	Package	Uncore U-box UCLK Fixed Counter
C10H	3088	MSR_U_PMON_EVNTSEL0	Package	Uncore U-box Perfmon Event Select for U-box Counter 0
C11H	3089	MSR_U_PMON_EVNTSEL1	Package	Uncore U-box Perfmon Event Select for U-box Counter 1
C16H	3094	MSR_U_PMON_CTR0	Package	Uncore U-box Perfmon Counter 0
C17H	3095	MSR_U_PMON_CTR1	Package	Uncore U-box Perfmon Counter 1
C24H	3108	MSR_PCU_PMON_BOX_CTL	Package	Uncore PCU Perfmon for PCU-box-wide Control
C30H	3120	MSR_PCU_PMON_EVNTSEL0	Package	Uncore PCU Perfmon Event Select for PCU Counter 0
C31H	3121	MSR_PCU_PMON_EVNTSEL1	Package	Uncore PCU Perfmon Event Select for PCU Counter 1
C32H	3122	MSR_PCU_PMON_EVNTSEL2	Package	Uncore PCU Perfmon Event Select for PCU Counter 2
C33H	3123	MSR_PCU_PMON_EVNTSEL3	Package	Uncore PCU Perfmon Event Select for PCU Counter 3
C34H	3124	MSR_PCU_PMON_BOX_FILTER	Package	Uncore PCU Perfmon box-wide Filter
C36H	3126	MSR_PCU_PMON_CTR0	Package	Uncore PCU Perfmon Counter 0
C37H	3127	MSR_PCU_PMON_CTR1	Package	Uncore PCU Perfmon Counter 1
C38H	3128	MSR_PCU_PMON_CTR2	Package	Uncore PCU Perfmon Counter 2
C39H	3129	MSR_PCU_PMON_CTR3	Package	Uncore PCU Perfmon Counter 3
D04H	3332	MSR_CO_PMON_BOX_CTL	Package	Uncore C-box O Perfmon Local Box Wide Control
D10H	3344	MSR_CO_PMON_EVNTSEL0	Package	Uncore C-box 0 Perfmon Event Select for C-box 0 Counter 0
D11H	3345	MSR_CO_PMON_EVNTSEL1	Package	Uncore C-box 0 Perfmon Event Select for C-box 0 Counter 1
D12H	3346	MSR_CO_PMON_EVNTSEL2	Package	Uncore C-box 0 Perfmon Event Select for C-box 0 Counter 2
D13H	3347	MSR_CO_PMON_EVNTSEL3	Package	Uncore C-box 0 Perfmon Event Select for C-box 0 Counter 3
D14H	3348	MSR_CO_PMON_BOX_FILTER	Package	Uncore C-box O Perfmon Box Wide Filter
D16H	3350	MSR_CO_PMON_CTRO	Package	Uncore C-box O Perfmon Counter O
D17H	3351	MSR_CO_PMON_CTR1	Package	Uncore C-box O Perfmon Counter 1
D18H	3352	MSR_CO_PMON_CTR2	Package	Uncore C-box O Perfmon Counter 2
D19H	3353	MSR_CO_PMON_CTR3	Package	Uncore C-box O Perfmon Counter 3
D24H	3364	MSR_C1_PMON_BOX_CTL	Package	Uncore C-box 1 Perfmon Local Box Wide Control

Table 2-24. Uncore PMU MSRs in Intel® Xeon® Processor E5 Family (Contd.)

Regi Add	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
D30H	3376	MSR_C1_PMON_EVNTSEL0	Package	Uncore C-box 1 Perfmon Event Select for C-box 1 Counter 0
D31H	3377	MSR_C1_PMON_EVNTSEL1	Package	Uncore C-box 1 Perfmon Event Select for C-box 1 Counter 1
D32H	3378	MSR_C1_PMON_EVNTSEL2	Package	Uncore C-box 1 Perfmon Event Select for C-box 1 Counter 2
D33H	3379	MSR_C1_PMON_EVNTSEL3	Package	Uncore C-box 1 Perfmon Event Select for C-box 1 Counter 3
D34H	3380	MSR_C1_PMON_BOX_FILTER	Package	Uncore C-box 1 Perfmon Box Wide Filter
D36H	3382	MSR_C1_PMON_CTR0	Package	Uncore C-box 1 Perfmon Counter 0
D37H	3383	MSR_C1_PMON_CTR1	Package	Uncore C-box 1 Perfmon Counter 1
D38H	3384	MSR_C1_PMON_CTR2	Package	Uncore C-box 1 Perfmon Counter 2
D39H	3385	MSR_C1_PMON_CTR3	Package	Uncore C-box 1 Perfmon Counter 3
D44H	3396	MSR_C2_PMON_BOX_CTL	Package	Uncore C-box 2 Perfmon Local Box Wide Control
D50H	3408	MSR_C2_PMON_EVNTSEL0	Package	Uncore C-box 2 Perfmon Event Select for C-box 2 Counter 0
D51H	3409	MSR_C2_PMON_EVNTSEL1	Package	Uncore C-box 2 Perfmon Event Select for C-box 2 Counter 1
D52H	3410	MSR_C2_PMON_EVNTSEL2	Package	Uncore C-box 2 Perfmon Event Select for C-box 2 Counter 2
D53H	3411	MSR_C2_PMON_EVNTSEL3	Package	Uncore C-box 2 Perfmon Event Select for C-box 2 Counter 3
D54H	3412	MSR_C2_PMON_BOX_FILTER	Package	Uncore C-box 2 Perfmon Box Wide Filter
D56H	3414	MSR_C2_PMON_CTR0	Package	Uncore C-box 2 Perfmon Counter 0
D57H	3415	MSR_C2_PMON_CTR1	Package	Uncore C-box 2 Perfmon Counter 1
D58H	3416	MSR_C2_PMON_CTR2	Package	Uncore C-box 2 Perfmon Counter 2
D59H	3417	MSR_C2_PMON_CTR3	Package	Uncore C-box 2 Perfmon Counter 3
D64H	3428	MSR_C3_PMON_BOX_CTL	Package	Uncore C-box 3 Perfmon Local Box Wide Control
D70H	3440	MSR_C3_PMON_EVNTSEL0	Package	Uncore C-box 3 Perfmon Event Select for C-box 3 Counter 0
D71H	3441	MSR_C3_PMON_EVNTSEL1	Package	Uncore C-box 3 Perfmon Event Select for C-box 3 Counter 1
D72H	3442	MSR_C3_PMON_EVNTSEL2	Package	Uncore C-box 3 Perfmon Event Select for C-box 3 Counter 2
D73H	3443	MSR_C3_PMON_EVNTSEL3	Package	Uncore C-box 3 Perfmon Event Select for C-box 3 Counter 3
D74H	3444	MSR_C3_PMON_BOX_FILTER	Package	Uncore C-box 3 Perfmon Box Wide Filter
D76H	3446	MSR_C3_PMON_CTR0	Package	Uncore C-box 3 Perfmon Counter 0
D77H	3447	MSR_C3_PMON_CTR1	Package	Uncore C-box 3 Perfmon Counter 1
D78H	3448	MSR_C3_PMON_CTR2	Package	Uncore C-box 3 Perfmon Counter 2
D79H	3449	MSR_C3_PMON_CTR3	Package	Uncore C-box 3 Perfmon Counter 3

Table 2-24. Uncore PMU MSRs in Intel® Xeon® Processor E5 Family (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
D84H	3460	MSR_C4_PMON_BOX_CTL	Package	Uncore C-box 4 Perfmon Local Box Wide Control
D90H	3472	MSR_C4_PMON_EVNTSEL0	Package	Uncore C-box 4 Perfmon Event Select for C-box 4 Counter 0
D91H	3473	MSR_C4_PMON_EVNTSEL1	Package	Uncore C-box 4 Perfmon Event Select for C-box 4 Counter 1
D92H	3474	MSR_C4_PMON_EVNTSEL2	Package	Uncore C-box 4 Perfmon Event Select for C-box 4 Counter 2
D93H	3475	MSR_C4_PMON_EVNTSEL3	Package	Uncore C-box 4 Perfmon Event Select for C-box 4 Counter 3
D94H	3476	MSR_C4_PMON_BOX_FILTER	Package	Uncore C-box 4 Perfmon Box Wide Filter
D96H	3478	MSR_C4_PMON_CTR0	Package	Uncore C-box 4 Perfmon Counter 0
D97H	3479	MSR_C4_PMON_CTR1	Package	Uncore C-box 4 Perfmon Counter 1
D98H	3480	MSR_C4_PMON_CTR2	Package	Uncore C-box 4 Perfmon Counter 2
D99H	3481	MSR_C4_PMON_CTR3	Package	Uncore C-box 4 Perfmon Counter 3
DA4H	3492	MSR_C5_PMON_BOX_CTL	Package	Uncore C-box 5 Perfmon Local Box Wide Control
DB0H	3504	MSR_C5_PMON_EVNTSEL0	Package	Uncore C-box 5 Perfmon Event Select for C-box 5 Counter 0
DB1H	3505	MSR_C5_PMON_EVNTSEL1	Package	Uncore C-box 5 Perfmon Event Select for C-box 5 Counter 1
DB2H	3506	MSR_C5_PMON_EVNTSEL2	Package	Uncore C-box 5 Perfmon Event Select for C-box 5 Counter 2
DB3H	3507	MSR_C5_PMON_EVNTSEL3	Package	Uncore C-box 5 Perfmon Event Select for C-box 5 Counter 3
DB4H	3508	MSR_C5_PMON_BOX_FILTER	Package	Uncore C-box 5 Perfmon Box Wide Filter
DB6H	3510	MSR_C5_PMON_CTR0	Package	Uncore C-box 5 Perfmon Counter 0
DB7H	3511	MSR_C5_PMON_CTR1	Package	Uncore C-box 5 Perfmon Counter 1
DB8H	3512	MSR_C5_PMON_CTR2	Package	Uncore C-box 5 Perfmon Counter 2
DB9H	3513	MSR_C5_PMON_CTR3	Package	Uncore C-box 5 Perfmon Counter 3
DC4H	3524	MSR_C6_PMON_BOX_CTL	Package	Uncore C-box 6 Perfmon Local Box Wide Control
DDOH	3536	MSR_C6_PMON_EVNTSEL0	Package	Uncore C-box 6 Perfmon Event Select for C-box 6 Counter 0
DD1H	3537	MSR_C6_PMON_EVNTSEL1	Package	Uncore C-box 6 Perfmon Event Select for C-box 6 Counter 1
DD2H	3538	MSR_C6_PMON_EVNTSEL2	Package	Uncore C-box 6 Perfmon Event Select for C-box 6 Counter 2
DD3H	3539	MSR_C6_PMON_EVNTSEL3	Package	Uncore C-box 6 Perfmon Event Select for C-box 6 Counter 3
DD4H	3540	MSR_C6_PMON_BOX_FILTER	Package	Uncore C-box 6 Perfmon Box Wide Filter
DD6H	3542	MSR_C6_PMON_CTR0	Package	Uncore C-box 6 Perfmon Counter 0
DD7H	3543	MSR_C6_PMON_CTR1	Package	Uncore C-box 6 Perfmon Counter 1
DD8H	3544	MSR_C6_PMON_CTR2	Package	Uncore C-box 6 Perfmon Counter 2

Table 2-24. Uncore PMU MSRs in Intel® Xeon® Processor E5 Family (Contd.)

Regi Add	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
DD9H	3545	MSR_C6_PMON_CTR3	Package	Uncore C-box 6 Perfmon Counter 3
DE4H	3556	MSR_C7_PMON_BOX_CTL	Package	Uncore C-box 7 Perfmon Local Box Wide Control
DF0H	3568	MSR_C7_PMON_EVNTSEL0	Package	Uncore C-box 7 Perfmon Event Select for C-box 7 Counter 0
DF1H	3569	MSR_C7_PMON_EVNTSEL1	Package	Uncore C-box 7 Perfmon Event Select for C-box 7 Counter 1
DF2H	3570	MSR_C7_PMON_EVNTSEL2	Package	Uncore C-box 7 Perfmon Event Select for C-box 7 Counter 2
DF3H	3571	MSR_C7_PMON_EVNTSEL3	Package	Uncore C-box 7 Perfmon Event Select for C-box 7 Counter 3
DF4H	3572	MSR_C7_PMON_BOX_FILTER	Package	Uncore C-box 7 Perfmon Box Wide Filter
DF6H	3574	MSR_C7_PMON_CTR0	Package	Uncore C-box 7 Perfmon Counter 0
DF7H	3575	MSR_C7_PMON_CTR1	Package	Uncore C-box 7 Perfmon Counter 1
DF8H	3576	MSR_C7_PMON_CTR2	Package	Uncore C-box 7 Perfmon Counter 2
DF9H	3577	MSR_C7_PMON_CTR3	Package	Uncore C-box 7 Perfmon Counter 3

2.12 MSRS IN THE 3RD GENERATION INTEL® CORE™ PROCESSOR FAMILY (BASED ON IVY BRIDGE MICROARCHITECTURE)

The 3rd generation Intel[®] Core[™] processor family and the Intel[®] Xeon[®] processor E3-1200v2 product family based on Ivy Bridge microarchitecture support the MSR interfaces listed in Table 2-20, Table 2-21, Table 2-22, and Table 2-25. These processors have a CPUID signature with DisplayFamily_DisplayModel of 06_3AH.

Table 2-25. Additional MSRs Supported by 3rd Generation Intel® Core™ Processors (Based on Ivy Bridge Microarchitecture)

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
CEH	206	MSR_PLATFORM_INFO	Package	Platform Information
				Contains power management and other model specific features enumeration. See http://biosbits.org.
		7:0		Reserved
		15:8	Package	Maximum Non-Turbo Ratio (R/O)
				This is the ratio of the frequency that invariant TSC runs at. Frequency = ratio * 100 MHz.
		27:16		Reserved
		28	Package	Programmable Ratio Limit for Turbo Mode (R/O)
				When set to 1, indicates that Programmable Ratio Limit for Turbo mode is enabled. When set to 0, indicates Programmable Ratio Limit for Turbo mode is disabled.

Table 2-25. Additional MSRs Supported by 3rd Generation Intel® Core™ Processors (Based on Ivy Bridge Microarchitecture) (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		29	Package	Programmable TDP Limit for Turbo Mode (R/O) When set to 1, indicates that TDP Limit for Turbo mode is programmable. When set to 0, indicates that TDP Limit for Turbo mode is not programmable.
		31:30		Reserved
		32	Package	Low Power Mode Support (LPM) (R/O) When set to 1, indicates that LPM is supported. When set to 0, indicates LPM is not supported.
		34:33	Package	Number of ConfigTDP Levels (R/O) 00: Only Base TDP level available. 01: One additional TDP level available. 02: Two additional TDP level available. 03: Reserved
		39:35		Reserved
		47:40	Package	Maximum Efficiency Ratio (R/O) This is the minimum ratio (maximum efficiency) that the processor can operate, in units of 100MHz.
		55:48	Package	Minimum Operating Ratio (R/O) Contains the minimum supported operating ratio in units of 100 MHz.
		63:56		Reserved
E2H	226	MSR_PKG_CST_CONFIG_CONTROL	Core	C-State Configuration Control (R/W) Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States. See http://biosbits.org.
		2:0		Package C-State Limit (R/W) Specifies the lowest processor-specific C-state code name (consuming the least power) for the package. The default is set as factory-configured package C-state limit. The following C-state code name encodings are supported: 000b: C0/C1 (no package C-sate support) 001b: C2 010b: C6 no retention 011b: C6 retention 100b: C7 101b: C7s 111: No package C-state limit. Note: This field cannot be used to limit package C-state to C3.

Table 2-25. Additional MSRs Supported by 3rd Generation Intel® Core™ Processors (Based on Ivy Bridge Microarchitecture) (Contd.)

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	1		
		9:3		Reserved
		10		I/O MWAIT Redirection Enable (R/W)
				When set, will map IO_read instructions sent to IO register specified by MSR_PMG_IO_CAPTURE_BASE to MWAIT instructions.
		14:11		Reserved
		15		CFG Lock (R/WO)
				When set, locks bits 15:0 of this register until next reset.
		24:16		Reserved
		25		C3 State Auto Demotion Enable (R/W)
				When set, the processor will conditionally demote C6/C7 requests to C3 based on uncore auto-demote information.
		26		C1 State Auto Demotion Enable (R/W)
				When set, the processor will conditionally demote C3/C6/C7 requests to C1 based on uncore autodemote information.
		27		Enable C3 Undemotion (R/W)
				When set, enables undemotion from demoted C3.
		28		Enable C1 Undemotion (R/W)
				When set, enables undemotion from demoted C1.
		63:29		Reserved
639H	1593	MSR_PPO_ENERGY_STATUS	Package	PPO Energy Status (R/O)
				See Section 15.10.4, "PPO/PP1 RAPL Domains."
648H	1608	MSR_CONFIG_TDP_NOMINAL	Package	Base TDP Ratio (R/O)
		7:0		Config_TDP_Base
				Base TDP level ratio to be used for this specific processor (in units of 100 MHz).
		63:8		Reserved
649H	1609	MSR_CONFIG_TDP_LEVEL1	Package	ConfigTDP Level 1 ratio and power level (R/O)
		14:0		PKG_TDP_LVL1
				Power setting for ConfigTDP Level 1.
		15		Reserved
		23:16		Config_TDP_LVL1_Ratio
				ConfigTDP level 1 ratio to be used for this specific processor.
		31:24		Reserved
		46:32		PKG_MAX_PWR_LVL1
				Max Power setting allowed for ConfigTDP Level 1.

Table 2-25. Additional MSRs Supported by 3rd Generation Intel® Core™ Processors (Based on Ivy Bridge Microarchitecture) (Contd.)

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		47		Reserved
		62:48		PKG_MIN_PWR_LVL1
				MIN Power setting allowed for ConfigTDP Level 1.
		63		Reserved
64AH	1610	MSR_CONFIG_TDP_LEVEL2	Package	ConfigTDP Level 2 ratio and power level (R/O)
		14:0		PKG_TDP_LVL2
				Power setting for ConfigTDP Level 2.
		15		Reserved
		23:16		Config_TDP_LVL2_Ratio
				ConfigTDP level 2 ratio to be used for this specific processor.
		31:24		Reserved
		46:32		PKG_MAX_PWR_LVL2
				Max Power setting allowed for ConfigTDP Level 2.
		47		Reserved
		62:48		PKG_MIN_PWR_LVL2
				MIN Power setting allowed for ConfigTDP Level 2.
		63		Reserved
64BH	1611	MSR_CONFIG_TDP_CONTROL	Package	ConfigTDP Control (R/W)
		1:0		TDP_LEVEL (RW/L)
				System BIOS can program this field.
		30:2		Reserved.
		31		Config_TDP_Lock (RW/L)
				When this bit is set, the content of this register is locked until a reset.
		63:32		Reserved
64CH	1612	MSR_TURBO_ACTIVATION_RATIO	Package	ConfigTDP Control (R/W)
		7:0		MAX_NON_TURBO_RATIO (RW/L)
				System BIOS can program this field.
		30:8		Reserved
		31		TURBO_ACTIVATION_RATIO_Lock (RW/L)
				When this bit is set, the content of this register is locked until a reset.
		63:32		Reserved
See Table	e 2-20, Ta	able 2-21, and Table 2-22 for other MSR	definitions applica	able to processors with CPUID signature 06_3AH.

2.12.1 MSRs In Intel[®] Xeon[®] Processor E5 v2 Product Family (Based on Ivy Bridge-E Microarchitecture)

Table 2-26 lists model-specific registers (MSRs) that are specific to the $Intel^{\circledR}$ Xeon $^{\circledR}$ Processor E5 v2 Product Family (based on Ivy Bridge-E microarchitecture). These processors have a CPUID signature with DisplayFamily_DisplayModel of 06_3EH, see Table 2-1. These processors supports the MSR interfaces listed in Table 2-20, and Table 2-26.

Table 2-26. MSRs Supported by Intel® Xeon® Processors E5 v2 Product Family (based on Ivy Bridge-E microarchitecture)

Regi Add	ister ress	Register Name / Bit Fields (Former MSR Name)	Scope	Bit Description
Hex	Dec			
4EH	78	IA32_PPIN_CTL (MSR_PPIN_CTL)	Package	Protected Processor Inventory Number Enable Control (R/W)
		0		LockOut (R/WO) See Table 2-2.
		1		Enable_PPIN (R/W) See Table 2-2.
		63:2		Reserved
4FH	79	IA32_PPIN (MSR_PPIN)	Package	Protected Processor Inventory Number (R/O)
		63:0		Protected Processor Inventory Number (R/O) See Table 2-2.
CEH	206	MSR_PLATFORM_INFO	Package	Platform Information
				Contains power management and other model specific features enumeration. See http://biosbits.org.
		7:0		Reserved
		15:8	Package	Maximum Non-Turbo Ratio (R/O)
				This is the ratio of the frequency that invariant TSC runs at. Frequency = ratio * 100 MHz.
		22:16		Reserved
		23	Package	PPIN_CAP (R/O)
				When set to 1, indicates that Protected Processor Inventory Number (PPIN) capability can be enabled for a privileged system inventory agent to read PPIN from MSR_PPIN.
				When set to 0, PPIN capability is not supported. An attempt to access MSR_PPIN_CTL or MSR_PPIN will cause #GP.
		27:24		Reserved
		28	Package	Programmable Ratio Limit for Turbo Mode (R/O)
				When set to 1, indicates that Programmable Ratio Limit for Turbo mode is enabled. When set to 0, indicates Programmable Ratio Limit for Turbo mode is disabled.

Table 2-26. MSRs Supported by Intel® Xeon® Processors E5 v2 Product Family (based on Ivy Bridge-E microarchitecture) (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
		29	Package	Programmable TDP Limit for Turbo Mode (R/0) When set to 1, indicates that TDP Limit for Turbo mode is programmable. When set to 0, indicates TDP Limit for Turbo mode is not programmable.
		30	Package	Programmable TJ OFFSET (R/O) When set to 1, indicates that MSR_TEMPERATURE_TARGET.[27:24] is valid and writable to specify a temperature offset.
		39:31		Reserved
		47:40	Package	Maximum Efficiency Ratio (R/O)
				This is the minimum ratio (maximum efficiency) that the processor can operate, in units of 100MHz.
		63:48		Reserved
E2H	226	MSR_PKG_CST_CONFIG_CONTROL	Core	C-State Configuration Control (R/W)
				Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-states.
				See http://biosbits.org.
		2:0		Package C-State Limit (R/W)
				Specifies the lowest processor-specific C-state code name (consuming the least power) for the package. The default is set as factory-configured package C-state limit.
				The following C-state code name encodings are supported:
				000b: CO/C1 (no package C-sate support)
				001b: C2
				010b: C6 no retention
				011b: C6 retention
				100b: C7
				101b: C7s
				111: No package C-state limit.
				Note: This field cannot be used to limit package C-state to C3.
		9:3		Reserved
		10		I/O MWAIT Redirection Enable (R/W)
				When set, will map IO_read instructions sent to IO register specified by MSR_PMG_IO_CAPTURE_BASE to MWAIT instructions.
		14:11		Reserved
		15		CFG Lock (R/WO)
				When set, locks bits 15:0 of this register until next reset.

Table 2-26. MSRs Supported by Intel® Xeon® Processors E5 v2 Product Family (based on Ivy Bridge-E microarchitecture) (Contd.)

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
		63:16		Reserved
179H	377	IA32_MCG_CAP	Thread	Global Machine Check Capability (R/O)
		7:0		Count
		8		MCG_CTL_P
		9		MCG_EXT_P
		10		MCP_CMCI_P
		11		MCG_TES_P
		15:12		Reserved
		23:16		MCG_EXT_CNT
		24		MCG_SER_P
		25		Reserved
		26		MCG_ELOG_P
		63:27		Reserved
17FH	383	MSR_ERROR_CONTROL	Package	MC Bank Error Configuration (R/W)
		0		Reserved
		1		MemError Log Enable (R/W)
				When set, enables IMC status bank to log additional info in bits 36:32.
		63:2		Reserved
1A2H	418	MSR_TEMPERATURE_TARGET	Package	Temperature Target
		15:0		Reserved
		23:16		Temperature Target (R/O)
				The minimum temperature at which PROCHOT# will be asserted. The value is degrees C.
		27:24		TCC Activation Offset (R/W)
				Specifies a temperature offset in degrees C from the temperature target (bits 23:16). PROCHOT# will assert at the offset target temperature. Write is permitted only if MSR_PLATFORM_INFO.[30] is set.
		63:28		Reserved
1AEH	430	MSR_TURBO_RATIO_LIMIT1	Package	Maximum Ratio Limit of Turbo Mode
				R/O if MSR_PLATFORM_INFO.[28] = 0.
				R/W if MSR_PLATFORM_INFO.[28] = 1.
		7:0	Package	Maximum Ratio Limit for 9C
				Maximum turbo ratio limit of 9 core active.
		15:8	Package	Maximum Ratio Limit for 10C
		22.10		Maximum turbo ratio limit of 10 core active.
		23:16	Package	Maximum Ratio Limit for 11C Maximum turbo ratio limit of 11 core active.
				riaxiiiiuiii turbo ratio iiiiiit OT 11 COFE active.

Table 2-26. MSRs Supported by Intel® Xeon® Processors E5 v2 Product Family (based on Ivy Bridge-E microarchitecture) (Contd.)

Register Address		Register Name / Bit Fields	Scope Scope	Bit Description
Hex	Dec	(Former MSR Name)		
		31:24	Package	Maximum Ratio Limit for 12C
				Maximum turbo ratio limit of 12 core active.
		63:32		Reserved
285H	645	IA32_MC5_CTL2	Package	See Table 2-2.
286H	646	IA32_MC6_CTL2	Package	See Table 2-2.
287H	647	IA32_MC7_CTL2	Package	See Table 2-2.
288H	648	IA32_MC8_CTL2	Package	See Table 2-2.
289H	649	IA32_MC9_CTL2	Package	See Table 2-2.
28AH	650	IA32_MC10_CTL2	Package	See Table 2-2.
28BH	651	IA32_MC11_CTL2	Package	See Table 2-2.
28CH	652	IA32_MC12_CTL2	Package	See Table 2-2.
28DH	653	IA32_MC13_CTL2	Package	See Table 2-2.
28EH	654	IA32_MC14_CTL2	Package	See Table 2-2.
28FH	655	IA32_MC15_CTL2	Package	See Table 2-2.
290H	656	IA32_MC16_CTL2	Package	See Table 2-2.
291H	657	IA32_MC17_CTL2	Package	See Table 2-2.
292H	658	IA32_MC18_CTL2	Package	See Table 2-2.
293H	659	IA32_MC19_CTL2	Package	See Table 2-2.
294H	660	IA32_MC20_CTL2	Package	See Table 2-2.
295H	661	IA32_MC21_CTL2	Package	See Table 2-2.
296H	662	IA32_MC22_CTL2	Package	See Table 2-2.
297H	663	IA32_MC23_CTL2	Package	See Table 2-2.
298H	664	IA32_MC24_CTL2	Package	See Table 2-2.
299H	665	IA32_MC25_CTL2	Package	See Table 2-2.
29AH	666	IA32_MC26_CTL2	Package	See Table 2-2.
29BH	667	IA32_MC27_CTL2	Package	See Table 2-2.
29CH	668	IA32_MC28_CTL2	Package	See Table 2-2.
414H	1044	IA32_MC5_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
415H	1045	IA32_MC5_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
416H	1046	IA32_MC5_ADDR	Package	Bank MC5 reports MC errors from the Intel QPI module.
417H	1047	IA32_MC5_MISC	Package	
418H	1048	IA32_MC6_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
419H	1049	IA32_MC6_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
41AH	1050	IA32_MC6_ADDR	Package	Bank MC6 reports MC errors from the integrated I/O module.
41BH	1051	IA32_MC6_MISC	Package	

Table 2-26. MSRs Supported by Intel® Xeon® Processors E5 v2 Product Family (based on Ivy Bridge-E microarchitecture) (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
41CH	1052	IA32_MC7_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
41DH	1053	IA32_MC7_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
41EH	1054	IA32_MC7_ADDR	Package	Banks MC7 and MC 8 report MC errors from the two home agents.
41FH	1055	IA32_MC7_MISC	Package	
420H	1056	IA32_MC8_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
421H	1057	IA32_MC8_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
422H	1058	IA32_MC8_ADDR	Package	Banks MC7 and MC 8 report MC errors from the two home agents.
423H	1059	IA32_MC8_MISC	Package	
424H	1060	IA32_MC9_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
425H	1061	IA32_MC9_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
426H	1062	IA32_MC9_ADDR	Package	Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.
427H	1063	IA32_MC9_MISC	Package	
428H	1064	IA32_MC10_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
429H	1065	IA32_MC10_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
42AH	1066	IA32_MC10_ADDR	Package	Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.
42BH	1067	IA32_MC10_MISC	Package	
42CH	1068	IA32_MC11_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
42DH	1069	IA32_MC11_STATUS	Package	Bank MC11 reports MC errors from a specific channel
42EH	1070	IA32_MC11_ADDR	Package	of the integrated memory controller.
42FH	1071	IA32_MC11_MISC	Package	
430H	1072	IA32_MC12_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
431H	1073	IA32_MC12_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
432H	1074	IA32_MC12_ADDR	Package	Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.
433H	1075	IA32_MC12_MISC	Package	
434H	1076	IA32_MC13_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
435H	1077	IA32_MC13_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
436H	1078	IA32_MC13_ADDR	Package	Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.
437H	1079	IA32_MC13_MISC	Package	
438H	1080	IA32_MC14_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
439H	1081	IA32_MC14_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
43AH	1082	IA32_MC14_ADDR	Package	Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.
43BH	1083	IA32_MC14_MISC	Package	

Table 2-26. MSRs Supported by Intel® Xeon® Processors E5 v2 Product Family (based on Ivy Bridge-E microarchitecture) (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
43CH	1084	IA32_MC15_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
43DH	1085	IA32_MC15_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
43EH	1086	IA32_MC15_ADDR	Package	Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.
43FH	1087	IA32_MC15_MISC	Package	
440H	1088	IA32_MC16_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
441H	1089	IA32_MC16_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
442H	1090	IA32_MC16_ADDR	Package	Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.
443H	1091	IA32_MC16_MISC	Package	
444H	1092	IA32_MC17_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
445H	1093	IA32_MC17_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
446H	1094	IA32_MC17_ADDR	Package	Bank MC17 reports MC errors from a specific CBo (core broadcast) and its corresponding slice of L3.
447H	1095	IA32_MC17_MISC	Package	
448H	1096	IA32_MC18_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
449H	1097	IA32_MC18_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
44AH	1098	IA32_MC18_ADDR	Package	Bank MC18 reports MC errors from a specific CBo (core broadcast) and its corresponding slice of L3.
44BH	1099	IA32_MC18_MISC	Package	
44CH	1100	IA32_MC19_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
44DH	1101	IA32_MC19_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
44EH	1102	IA32_MC19_ADDR	Package	Bank MC19 reports MC errors from a specific CBo (core broadcast) and its corresponding slice of L3.
44FH	1103	IA32_MC19_MISC	Package	
450H	1104	IA32_MC20_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
451H	1105	IA32_MC20_STATUS	Package	Bank MC20 reports MC errors from a specific CBo (core
452H	1106	IA32_MC20_ADDR	Package	broadcast) and its corresponding slice of L3.
453H	1107	IA32_MC20_MISC	Package	
454H	1108	IA32_MC21_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
455H	1109	IA32_MC21_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
456H	1110	IA32_MC21_ADDR	Package	Bank MC21 reports MC errors from a specific CBo (core broadcast) and its corresponding slice of L3.
457H	1111	IA32_MC21_MISC	Package	
458H	1112	IA32_MC22_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
459H	1113	IA32_MC22_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
45AH	1114	IA32_MC22_ADDR	Package	Bank MC22 reports MC errors from a specific CBo (core broadcast) and its corresponding slice of L3.
45BH	1115	IA32_MC22_MISC	Package	

Table 2-26. MSRs Supported by Intel® Xeon® Processors E5 v2 Product Family (based on Ivy Bridge-E microarchitecture) (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
45CH	1116	IA32_MC23_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
45DH	1117	IA32_MC23_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
45EH	1118	IA32_MC23_ADDR	Package	Bank MC23 reports MC errors from a specific CBo (core broadcast) and its corresponding slice of L3.
45FH	1119	IA32_MC23_MISC	Package	, , ,
460H	1120	IA32_MC24_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
461H	1121	IA32_MC24_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
462H	1122	IA32_MC24_ADDR	Package	Bank MC24 reports MC errors from a specific CBo (core broadcast) and its corresponding slice of L3.
463H	1123	IA32_MC24_MISC	Package	
464H	1124	IA32_MC25_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
465H	1125	IA32_MC25_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
466H	1126	IA32_MC25_ADDR	Package	Bank MC25 reports MC errors from a specific CBo (core broadcast) and its corresponding slice of L3.
467H	1127	IA32_MC2MISC	Package	
468H	1128	IA32_MC26_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
469H	1129	IA32_MC26_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
46AH	1130	IA32_MC26_ADDR	Package	Bank MC26 reports MC errors from a specific CBo (core broadcast) and its corresponding slice of L3.
46BH	1131	IA32_MC26_MISC	Package	7 , , ,
46CH	1132	IA32_MC27_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
46DH	1133	IA32_MC27_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
46EH	1134	IA32_MC27_ADDR	Package	Bank MC27 reports MC errors from a specific CBo (core broadcast) and its corresponding slice of L3.
46FH	1135	IA32_MC27_MISC	Package	, , ,
470H	1136	IA32_MC28_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
471H	1137	IA32_MC28_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
472H	1138	IA32_MC28_ADDR	Package	Bank MC28 reports MC errors from a specific CBo (core broadcast) and its corresponding slice of L3.
473H	1139	IA32_MC28_MISC	Package	, , ,
613H	1555	MSR_PKG_PERF_STATUS	Package	Package RAPL Perf Status (R/O)
618H	1560	MSR_DRAM_POWER_LIMIT	Package	DRAM RAPL Power Limit Control (R/W)
				See Section 15.10.5, "DRAM RAPL Domain."
619H	1561	MSR_DRAM_ENERGY_STATUS	Package	DRAM Energy Status (R/O)
				See Section 15.10.5, "DRAM RAPL Domain."
61BH	1563	MSR_DRAM_PERF_STATUS	Package	DRAM Performance Throttling Status (R/O)
64.611	1564	MCD DDAM DOLIED INICO		See Section 15.10.5, "DRAM RAPL Domain."
61CH	1564	MSR_DRAM_POWER_INFO	Package	DRAM RAPL Parameters (R/W) See Section 15.10.5, "DRAM RAPL Domain."
639H	1593	MSR_PPO_ENERGY_STATUS	Package	PPO Energy Status (R/O)
03311	1333	TISK_ITO_CINCKOT_STATOS	1 dekage	See Section 15.10.4, "PPO/PP1 RAPL Domains."
See Table	e 2-20, fo	ı r other MSR definitions applicable to Int	tel Xeon processo	

2.12.2 Additional MSRs Supported by Intel® Xeon® Processor E7 v2 Family

Intel $^{(8)}$ Xeon $^{(8)}$ processor E7 v2 family (based on Ivy Bridge-E microarchitecture) with CPUID DisplayFamily_DisplayModel signature 06_3EH supports the MSR interfaces listed in Table 2-20, Table 2-26, and Table 2-27.

Table 2-27. Additional MSRs Supported by Intel® Xeon® Processor E7 v2 Family with DisplayFamily_DisplayModel Signature 06_3EH

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
ЗАН	58	IA32_FEATURE_CONTROL	Thread	Control Features in Intel 64 Processor (R/W)
				See Table 2-2.
		0		Lock (R/WL)
		1		Enable VMX Inside SMX Operation (R/WL)
		2		Enable VMX Outside SMX Operation (R/WL)
		14:8		SENTER Local Functions Enables (R/WL)
		15		SENTER Global Functions Enable (R/WL)
		63:16		Reserved
179H	377	IA32_MCG_CAP	Thread	Global Machine Check Capability (R/O)
		7:0		Count
		8		MCG_CTL_P
		9		MCG_EXT_P
		10		MCP_CMCI_P
		11		MCG_TES_P
		15:12		Reserved
		23:16		MCG_EXT_CNT
		24		MCG_SER_P
		63:25		Reserved
17AH	378	IA32_MCG_STATUS	Thread	Global Machine Check Status (R/WO)
		0		RIPV
		1		EIPV
		2		MCIP
		3		LMCE Signaled
		63:4		Reserved
1AEH	430	MSR_TURBO_RATIO_LIMIT1	Package	Maximum Ratio Limit of Turbo Mode
				R/O if MSR_PLATFORM_INFO.[28] = 0.
				R/W if MSR_PLATFORM_INFO.[28] = 1.
		7:0	Package	Maximum Ratio Limit for 9C
				Maximum turbo ratio limit of 9 core active.
		15:8	Package	Maximum Ratio Limit for 10C
				Maximum turbo ratio limit of 10core active.

Table 2-27. Additional MSRs Supported by Intel® Xeon® Processor E7 v2 Family with DisplayFamily_DisplayModel Signature 06_3EH

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		23:16	Package	Maximum Ratio Limit for 11C Maximum turbo ratio limit of 11 core active.
		31:24	Package	Maximum Ratio Limit for 12C Maximum turbo ratio limit of 12 core active.
		39:32	Package	Maximum Ratio Limit for 13C Maximum turbo ratio limit of 13 core active.
		47:40	Package	Maximum Ratio Limit for 14C Maximum turbo ratio limit of 14 core active.
		55:48	Package	Maximum Ratio Limit for 15C Maximum turbo ratio limit of 15 core active.
		62:56		Reserved
		63	Package	Semaphore for Turbo Ratio Limit Configuration
				If 1, the processor uses override configuration specified in MSR_TURBO_RATIO_LIMIT and MSR_TURBO_RATIO_LIMIT1.
				If 0, the processor uses factory-set configuration (Default).
29DH	669	IA32_MC29_CTL2	Package	See Table 2-2.
29EH	670	IA32_MC30_CTL2	Package	See Table 2-2.
29FH	671	IA32_MC31_CTL2	Package	See Table 2-2.
3F1H	1009	IA32_PEBS_ENABLE (MSR_PEBS_ENABLE)	Thread	See Section 20.3.1.1.1, "Processor Event Based Sampling (PEBS)."
		n:0		Enable PEBS on IA32_PMCx. (R/W)
		31:n+1		Reserved
		32+ <i>m</i> :32		Enable Load Latency on IA32_PMCx. (R/W)
		63:33+m		Reserved
41BH	1051	IA32_MC6_MISC	Package	Misc MAC Information of Integrated I/O (R/O) See Section 16.3.2.4.
		5:0		Recoverable Address LSB
		8:6		Address Mode
		15:9		Reserved
		31:16		PCI Express Requestor ID
		39:32		PCI Express Segment Number
		63:32		Reserved
474H	1140	IA32_MC29_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
475H	1141	IA32_MC29_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
476H	1142	IA32_MC29_ADDR	Package	Bank MC29 reports MC errors from a specific CBo (cor broadcast) and its corresponding slice of L3.
477H	1143	IA32_MC29_MISC	Package	

Table 2-27. Additional MSRs Supported by Intel® Xeon® Processor E7 v2 Family with DisplayFamily_DisplayModel Signature 06_3EH

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
478H	1144	IA32_MC30_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
479H	1145	IA32_MC30_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
47AH	1146	IA32_MC30_ADDR	Package	Bank MC30 reports MC errors from a specific CBo (core broadcast) and its corresponding slice of L3.
47BH	1147	IA32_MC30_MISC	Package	, , ,
47CH	1148	IA32_MC31_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
47DH	1149	IA32_MC31_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
47EH	1150	IA32_MC31_ADDR	Package	Bank MC31 reports MC errors from a specific CBo (core broadcast) and its corresponding slice of L3.
47FH	1147	IA32_MC31_MISC	Package	, , ,
See Table	e 2-20, Ta	ble 2-26 for other MSR definitions applical	ble to Intel Xe	on processor E7 v2 with CPUID signature 06_3AH.

NOTES:

2.12.3 Additional Uncore PMU MSRs in the Intel® Xeon® Processor E5 v2 and E7 v2 Families

Intel Xeon Processor E5 v2 and E7 v2 families are based on the Ivy Bridge-E microarchitecture. The MSR-based uncore PMU interfaces are listed in Table 2-24 and Table 2-28. For complete detail of the uncore PMU, refer to Intel Xeon Processor E5 v2 Product Family Uncore Performance Monitoring Guide. These processors have a CPUID signature with DisplayFamily_DisplayModel of 06_3EH.

Table 2-28. Uncore PMU MSRs in Intel® Xeon® Processor E5 v2 and E7 v2 Families

_	ster ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
COOH	3072	MSR_PMON_GLOBAL_CTL	Package	Uncore Perfmon Per-Socket Global Control
CO1H	3073	MSR_PMON_GLOBAL_STATUS	Package	Uncore Perfmon Per-Socket Global Status
C06H	3078	MSR_PMON_GLOBAL_CONFIG	Package	Uncore Perfmon Per-Socket Global Configuration
C15H	3093	MSR_U_PMON_BOX_STATUS	Package	Uncore U-box Perfmon U-Box Wide Status
C35H	3125	MSR_PCU_PMON_BOX_STATUS	Package	Uncore PCU Perfmon Box Wide Status
D1AH	3354	MSR_CO_PMON_BOX_FILTER1	Package	Uncore C-Box 0 Perfmon Box Wide Filter1
D3AH	3386	MSR_C1_PMON_BOX_FILTER1	Package	Uncore C-Box 1 Perfmon Box Wide Filter1
D5AH	3418	MSR_C2_PMON_BOX_FILTER1	Package	Uncore C-Box 2 Perfmon Box Wide Filter1
D7AH	3450	MSR_C3_PMON_BOX_FILTER1	Package	Uncore C-Box 3 Perfmon Box Wide Filter1
D9AH	3482	MSR_C4_PMON_BOX_FILTER1	Package	Uncore C-Box 4 Perfmon Box Wide Filter1
DBAH	3514	MSR_C5_PMON_BOX_FILTER1	Package	Uncore C-Box 5 Perfmon Box Wide Filter1
DDAH	3546	MSR_C6_PMON_BOX_FILTER1	Package	Uncore C-Box 6 Perfmon Box Wide Filter1
DFAH	3578	MSR_C7_PMON_BOX_FILTER1	Package	Uncore C-Box 7 Perfmon Box Wide Filter1
E04H	3588	MSR_C8_PMON_BOX_CTL	Package	Uncore C-Box 8 Perfmon Local Box Wide Control

^{1.} An override configuration lower than the factory-set configuration is always supported. An override configuration higher than the factory-set configuration is dependent on features specific to the processor and the platform.

Table 2-28. Uncore PMU MSRs in Intel® Xeon® Processor E5 v2 and E7 v2 Families (Contd.)

Regi Add	ister	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
E10H	3600	MSR_C8_PMON_EVNTSEL0	Package	Uncore C-Box 8 Perfmon Event Select for C-Box 8 Counter 0
E11H	3601	MSR_C8_PMON_EVNTSEL1	Package	Uncore C-Box 8 Perfmon Event Select for C-Box 8 Counter 1
E12H	3602	MSR_C8_PMON_EVNTSEL2	Package	Uncore C-Box 8 Perfmon Event Select for C-Box 8 Counter 2
E13H	3603	MSR_C8_PMON_EVNTSEL3	Package	Uncore C-Box 8 Perfmon Event Select for C-Box 8 Counter 3
E14H	3604	MSR_C8_PMON_BOX_FILTER	Package	Uncore C-Box 8 Perfmon Box Wide Filter
E16H	3606	MSR_C8_PMON_CTR0	Package	Uncore C-Box 8 Perfmon Counter 0
E17H	3607	MSR_C8_PMON_CTR1	Package	Uncore C-Box 8 Perfmon Counter 1
E18H	3608	MSR_C8_PMON_CTR2	Package	Uncore C-Box 8 Perfmon Counter 2
E19H	3609	MSR_C8_PMON_CTR3	Package	Uncore C-Box 8 Perfmon Counter 3
E1AH	3610	MSR_C8_PMON_BOX_FILTER1	Package	Uncore C-Box 8 Perfmon Box Wide Filter1
E24H	3620	MSR_C9_PMON_BOX_CTL	Package	Uncore C-Box 9 Perfmon Local Box Wide Control
E30H	3632	MSR_C9_PMON_EVNTSEL0	Package	Uncore C-Box 9 Perfmon Event Select for C-box 9 Counter 0
E31H	3633	MSR_C9_PMON_EVNTSEL1	Package	Uncore C-Box 9 Perfmon Event Select for C-box 9 Counter 1
E32H	3634	MSR_C9_PMON_EVNTSEL2	Package	Uncore C-Box 9 Perfmon Event Select for C-box 9 Counter 2
E33H	3635	MSR_C9_PMON_EVNTSEL3	Package	Uncore C-Box 9 Perfmon Event Select for C-box 9 Counter 3
E34H	3636	MSR_C9_PMON_BOX_FILTER	Package	Uncore C-Box 9 Perfmon Box Wide Filter
E36H	3638	MSR_C9_PMON_CTR0	Package	Uncore C-Box 9 Perfmon Counter 0
E37H	3639	MSR_C9_PMON_CTR1	Package	Uncore C-Box 9 Perfmon Counter 1
E38H	3640	MSR_C9_PMON_CTR2	Package	Uncore C-Box 9 Perfmon Counter 2
E39H	3641	MSR_C9_PMON_CTR3	Package	Uncore C-Box 9 Perfmon Counter 3
ЕЗАН	3642	MSR_C9_PMON_BOX_FILTER1	Package	Uncore C-Box 9 Perfmon Box Wide Filter1
E44H	3652	MSR_C10_PMON_BOX_CTL	Package	Uncore C-Box 10 Perfmon Local Box Wide Control
E50H	3664	MSR_C10_PMON_EVNTSEL0	Package	Uncore C-Box 10 Perfmon Event Select for C-Box 10 Counter 0
E51H	3665	MSR_C10_PMON_EVNTSEL1	Package	Uncore C-Box 10 Perfmon Event Select for C-Box 10 Counter 1
E52H	3666	MSR_C10_PMON_EVNTSEL2	Package	Uncore C-Box 10 Perfmon Event Select for C-Box 10 Counter 2
E53H	3667	MSR_C10_PMON_EVNTSEL3	Package	Uncore C-Box 10 Perfmon Event Select for C-Box 10 Counter 3
E54H	3668	MSR_C10_PMON_BOX_FILTER	Package	Uncore C-Box 10 Perfmon Box Wide Filter
E56H	3670	MSR_C10_PMON_CTR0	Package	Uncore C-Box 10 Perfmon Counter 0
E57H	3671	MSR_C10_PMON_CTR1	Package	Uncore C-Box 10 Perfmon Counter 1

Table 2-28. Uncore PMU MSRs in Intel® Xeon® Processor E5 v2 and E7 v2 Families (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
E58H	3672	MSR_C10_PMON_CTR2	Package	Uncore C-Box 10 Perfmon Counter 2
E59H	3673	MSR_C10_PMON_CTR3	Package	Uncore C-Box 10 Perfmon Counter 3
E5AH	3674	MSR_C10_PMON_BOX_FILTER1	Package	Uncore C-Box 10 Perfmon Box Wide Filter1
E64H	3684	MSR_C11_PMON_BOX_CTL	Package	Uncore C-Box 11 Perfmon Local Box Wide Control
E70H	3696	MSR_C11_PMON_EVNTSEL0	Package	Uncore C-Box 11 Perfmon Event Select for C-Box 11 Counter 0
E71H	3697	MSR_C11_PMON_EVNTSEL1	Package	Uncore C-Box 11 Perfmon Event Select for C-Box 11 Counter 1
E72H	3698	MSR_C11_PMON_EVNTSEL2	Package	Uncore C-Box 11 Perfmon Event Select for C-Box 11 Counter 2
E73H	3699	MSR_C11_PMON_EVNTSEL3	Package	Uncore C-Box 11 Perfmon Event Select for C-Box 11 Counter 3
E74H	3700	MSR_C11_PMON_BOX_FILTER	Package	Uncore C-Box 11 Perfmon Box Wide Filter
E76H	3702	MSR_C11_PMON_CTR0	Package	Uncore C-Box 11 Perfmon Counter 0
E77H	3703	MSR_C11_PMON_CTR1	Package	Uncore C-Box 11 Perfmon Counter 1
E78H	3704	MSR_C11_PMON_CTR2	Package	Uncore C-Box 11 Perfmon Counter 2
E79H	3705	MSR_C11_PMON_CTR3	Package	Uncore C-Box 11 Perfmon Counter 3
E7AH	3706	MSR_C11_PMON_BOX_FILTER1	Package	Uncore C-Box 11 Perfmon Box Wide Filter1
E84H	3716	MSR_C12_PMON_BOX_CTL	Package	Uncore C-Box 12 Perfmon Local Box Wide Control
E90H	3728	MSR_C12_PMON_EVNTSEL0	Package	Uncore C-Box 12 Perfmon Event Select for C-Box 12 Counter 0
E91H	3729	MSR_C12_PMON_EVNTSEL1	Package	Uncore C-Box 12 Perfmon Event Select for C-Box 12 Counter 1
E92H	3730	MSR_C12_PMON_EVNTSEL2	Package	Uncore C-Box 12 Perfmon Event Select for C-Box 12 Counter 2
E93H	3731	MSR_C12_PMON_EVNTSEL3	Package	Uncore C-Box 12 Perfmon Event Select for C-Box 12 Counter 3
E94H	3732	MSR_C12_PMON_BOX_FILTER	Package	Uncore C-Box 12 Perfmon Box Wide Filter
E96H	3734	MSR_C12_PMON_CTR0	Package	Uncore C-Box 12 Perfmon Counter 0
E97H	3735	MSR_C12_PMON_CTR1	Package	Uncore C-Box 12 Perfmon Counter 1
E98H	3736	MSR_C12_PMON_CTR2	Package	Uncore C-Box 12 Perfmon Counter 2
E99H	3737	MSR_C12_PMON_CTR3	Package	Uncore C-Box 12 Perfmon Counter 3
E9AH	3738	MSR_C12_PMON_BOX_FILTER1	Package	Uncore C-Box 12 Perfmon Box Wide Filter1
EA4H	3748	MSR_C13_PMON_BOX_CTL	Package	Uncore C-Box 13 Perfmon Local Box Wide Control
EB0H	3760	MSR_C13_PMON_EVNTSEL0	Package	Uncore C-Box 13 Perfmon Event Select for C-Box 13 Counter 0
EB1H	3761	MSR_C13_PMON_EVNTSEL1	Package	Uncore C-Box 13 Perfmon Event Select for C-Box 13 Counter 1
EB2H	3762	MSR_C13_PMON_EVNTSEL2	Package	Uncore C-Box 13 Perfmon Event Select for C-Box 13 Counter 2

Table 2-28. Uncore PMU MSRs in Intel® Xeon® Processor E5 v2 and E7 v2 Families (Contd.)

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
EB3H	3763	MSR_C13_PMON_EVNTSEL3	Package	Uncore C-Box 13 Perfmon Event Select for C-Box 13 Counter 3
EB4H	3764	MSR_C13_PMON_BOX_FILTER	Package	Uncore C-Box 13 Perfmon Box Wide Filter
EB6H	3766	MSR_C13_PMON_CTR0	Package	Uncore C-Box 13 Perfmon Counter 0
EB7H	3767	MSR_C13_PMON_CTR1	Package	Uncore C-Box 13 Perfmon Counter 1
EB8H	3768	MSR_C13_PMON_CTR2	Package	Uncore C-Box 13 Perfmon Counter 2
EB9H	3769	MSR_C13_PMON_CTR3	Package	Uncore C-Box 13 Perfmon Counter 3
EBAH	3770	MSR_C13_PMON_BOX_FILTER1	Package	Uncore C-Box 13 Perfmon Box Wide Filter1
EC4H	3780	MSR_C14_PMON_BOX_CTL	Package	Uncore C-Box 14 Perfmon Local Box Wide Control
EDOH	3792	MSR_C14_PMON_EVNTSEL0	Package	Uncore C-Box 14 Perfmon Event Select for C-Box 14 Counter 0
ED1H	3793	MSR_C14_PMON_EVNTSEL1	Package	Uncore C-Box 14 Perfmon Event Select for C-Box 14 Counter 1
ED2H	3794	MSR_C14_PMON_EVNTSEL2	Package	Uncore C-Box 14 Perfmon Event Select for C-Box 14 Counter 2
ED3H	3795	MSR_C14_PMON_EVNTSEL3	Package	Uncore C-Box 14 Perfmon Event Select for C-Box 14 Counter 3
ED4H	3796	MSR_C14_PMON_BOX_FILTER	Package	Uncore C-Box 14 Perfmon Box Wide Filter
ED6H	3798	MSR_C14_PMON_CTR0	Package	Uncore C-Box 14 Perfmon Counter 0
ED7H	3799	MSR_C14_PMON_CTR1	Package	Uncore C-Box 14 Perfmon Counter 1
ED8H	3800	MSR_C14_PMON_CTR2	Package	Uncore C-Box 14 Perfmon Counter 2
ED9H	3801	MSR_C14_PMON_CTR3	Package	Uncore C-Box 14 Perfmon Counter 3
EDAH	3802	MSR_C14_PMON_BOX_FILTER1	Package	Uncore C-Box 14 Perfmon Box Wide Filter1

2.13 MSRS IN THE 4TH GENERATION INTEL® CORE™ PROCESSORS (BASED ON HASWELL MICROARCHITECTURE)

The 4th generation Intel[®] Core[™] processor family and Intel[®] Xeon[®] processor E3-1200v3 product family (based on Haswell microarchitecture), with CPUID DisplayFamily_DisplayModel signature 06_3CH/06_45H/06_46H, support the MSR interfaces listed in Table 2-20, Table 2-21, Table 2-22, and Table 2-29. For an MSR listed in Table 2-20 that also appears in Table 2-29, Table 2-29 supersede Table 2-20.

The MSRs listed in Table 2-29 also apply to processors based on Haswell-E microarchitecture (see Section 2.14).

Table 2-29. Additional MSRs Supported by Processors based on the Haswell or Haswell-E microarchitectures

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
3BH	59	IA32_TSC_ADJUST	Thread	Per-Logical-Processor TSC ADJUST (R/W) See Table 2-2.

Table 2-29. Additional MSRs Supported by Processors based on the Haswell or Haswell-E microarchitectures

Regi Add	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	7		
CEH	206	MSR_PLATFORM_INFO	Package	Platform Information Contains power management and other model specific features enumeration. See http://biosbits.org.
		7:0		Reserved
		15:8	Package	Maximum Non-Turbo Ratio (R/O) This is the ratio of the frequency that invariant TSC runs at. Frequency = ratio * 100 MHz.
		27:16		Reserved
		28	Package	Programmable Ratio Limit for Turbo Mode (R/O) When set to 1, indicates that Programmable Ratio Limit for Turbo mode is enabled. When set to 0, indicates Programmable Ratio Limit for Turbo mode is disabled.
		29	Package	Programmable TDP Limit for Turbo Mode (R/O) When set to 1, indicates that TDP Limit for Turbo mode is programmable. When set to 0, indicates TDP Limit for Turbo mode is not programmable.
		31:30		Reserved
		32	Package	Low Power Mode Support (LPM) (R/O) When set to 1, indicates that LPM is supported. When set to 0, indicates LPM is not supported.
		34:33	Package	Number of ConfigTDP Levels (R/O) 00: Only Base TDP level available. 01: One additional TDP level available. 02: Two additional TDP level available. 03: Reserved
		39:35		Reserved
		47:40	Package	Maximum Efficiency Ratio (R/O) This is the minimum ratio (maximum efficiency) that the processor can operate, in units of 100MHz.
		55:48	Package	Minimum Operating Ratio (R/O) Contains the minimum supported operating ratio in units of 100 MHz.
		63:56		Reserved
186H	390	IA32_PERFEVTSEL0	Thread	Performance Event Select for Counter 0 (R/W) Supports all fields described inTable 2-2 and the fields below.
		32		IN_TX: See Section 20.3.6.5.1. When IN_TX (bit 32) is set, AnyThread (bit 21) should be cleared to prevent incorrect results.

Table 2-29. Additional MSRs Supported by Processors based on the Haswell or Haswell-E microarchitectures

Regi	ister ress	Register Name / Bit Fields Sco	Scope	Bit Description
Hex	Dec			
187H	391	IA32_PERFEVTSEL1	Thread	Performance Event Select for Counter 1 (R/W) Supports all fields described inTable 2-2 and the fields below.
		32		IN_TX: See Section 20.3.6.5.1. When IN_TX (bit 32) is set, AnyThread (bit 21) should be cleared to prevent incorrect results.
188H	392	IA32_PERFEVTSEL2	Thread	Performance Event Select for Counter 2 (R/W) Supports all fields described inTable 2-2 and the fields below.
		32		IN_TX: See Section 20.3.6.5.1. When IN_TX (bit 32) is set, AnyThread (bit 21) should be cleared to prevent incorrect results.
		33		IN_TXCP: See Section 20.3.6.5.1. When IN_TXCP=1 & IN_TX=1 and in sampling, a spurious PMI may occur and transactions may continuously abort near overflow conditions. Software should favor using IN_TXCP for counting over sampling. If sampling, software should use large "sample-after" value after clearing the counter configured to use IN_TXCP and also always reset the counter even when no overflow condition was reported.
189H	393	IA32_PERFEVTSEL3	Thread	Performance Event Select for Counter 3 (R/W) Supports all fields described inTable 2-2 and the fields below.
		32		IN_TX: See Section 20.3.6.5.1 When IN_TX (bit 32) is set, AnyThread (bit 21) should be cleared to prevent incorrect results.
1C8H	456	MSR_LBR_SELECT	Thread	Last Branch Record Filtering Select Register (R/W)
		0		CPL_EQ_0
		1		CPL_NEQ_0
		2		JCC
		3		NEAR_REL_CALL
		4		NEAR_IND_CALL
		5		NEAR_RET
		6		NEAR_IND_JMP
		7		NEAR_REL_JMP
		8		FAR_BRANCH
		9		EN_CALL_STACK
		63:9		Reserved
1D9H	473	IA32_DEBUGCTL	Thread	Debug Control (R/W) See Table 2-2.

Table 2-29. Additional MSRs Supported by Processors based on the Haswell or Haswell-E microarchitectures

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		0		LBR: Last Branch Record
		1		BTF
		5:2		Reserved
		6		TR: Branch Trace
		7		BTS: Log Branch Trace Message to BTS Buffer
		8		BTINT
		9		BTS_OFF_OS
		10		BTS_OFF_USER
		11		FREEZE_LBR_ON_PMI
		12		FREEZE_PERFMON_ON_PMI
		13		ENABLE_UNCORE_PMI
		14		FREEZE_WHILE_SMM
		15		RTM_DEBUG
		63:15		Reserved
491H	1169	IA32_VMX_VMFUNC	Thread	Capability Reporting Register of VM-Function Controls (R/O)
				See Table 2-2.
60BH	1548	MSR_PKGC_IRTL1	Package	Package C6/C7 Interrupt Response Limit 1 (R/W) This MSR defines the interrupt response time limit used by the processor to manage a transition to a package C6 or C7 state. The latency programmed in this register is for the shorter-latency sub C-states used by an MWAIT hint to a C6 or C7 state. Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		9:0		Interrupt Response Time Limit (R/W) Specifies the limit that should be used to decide if the package should be put into a package C6 or C7 state.
		12:10		Time Unit (R/W) Specifies the encoding value of time unit of the interrupt response time limit. See Table 2-20 for supported time unit encodings.
		14:13		Reserved
		15		Valid (R/W) Indicates whether the values in bits 12:0 are valid and can be used by the processor for package C-sate management.
		63:16		Reserved

Table 2-29. Additional MSRs Supported by Processors based on the Haswell or Haswell-E microarchitectures

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
60CH	1548	MSR_PKGC_IRTL2	Package	Package C6/C7 Interrupt Response Limit 2 (R/W) This MSR defines the interrupt response time limit used by the processor to manage a transition to a package C6 or C7 state. The latency programmed in this register is for the longer-latency sub C-states used by an MWAIT hint to a C6 or C7 state. Note: C-state values are processor specific C-state
				code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		9:0		Interrupt response time limit (R/W)
				Specifies the limit that should be used to decide if the package should be put into a package C6 or C7 state.
		12:10		Time Unit (R/W)
				Specifies the encoding value of time unit of the interrupt response time limit. See Table 2-20 for supported time unit encodings.
		14:13		Reserved
		15		Valid (R/W)
				Indicates whether the values in bits 12:0 are valid and can be used by the processor for package C-sate management.
		63:16		Reserved
613H	1555	MSR_PKG_PERF_STATUS	Package	PKG Perf Status (R/O)
				See Section 15.10.3, "Package RAPL Domain."
619H	1561	MSR_DRAM_ENERGY_STATUS	Package	DRAM Energy Status (R/O)
				See Section 15.10.5, "DRAM RAPL Domain."
61BH	1563	MSR_DRAM_PERF_STATUS	Package	DRAM Performance Throttling Status (R/O) See Section 15.10.5, "DRAM RAPL Domain."
648H	1608	MSR_CONFIG_TDP_NOMINAL	Package	Base TDP Ratio (R/O)
		7:0		Config_TDP_Base
				Base TDP level ratio to be used for this specific processor (in units of 100 MHz).
		63:8		Reserved
649H	1609	MSR_CONFIG_TDP_LEVEL1	Package	ConfigTDP Level 1 Ratio and Power Level (R/O)
		14:0		PKG_TDP_LVL1
				Power setting for ConfigTDP Level 1.
		15		Reserved
		23:16		Config_TDP_LVL1_Ratio
				ConfigTDP level 1 ratio to be used for this specific processor.
		31:24		Reserved

Table 2-29. Additional MSRs Supported by Processors based on the Haswell or Haswell-E microarchitectures

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		46:32		PKG_MAX_PWR_LVL1
				Max Power setting allowed for ConfigTDP Level 1.
		62:47		PKG_MIN_PWR_LVL1
				MIN Power setting allowed for ConfigTDP Level 1.
		63		Reserved
64AH	1610	MSR_CONFIG_TDP_LEVEL2	Package	ConfigTDP Level 2 Ratio and Power Level (R/O)
		14:0		PKG_TDP_LVL2
				Power setting for ConfigTDP Level 2.
		15		Reserved
		23:16		Config_TDP_LVL2_Ratio
				ConfigTDP level 2 ratio to be used for this specific processor.
		31:24		Reserved
		46:32		PKG_MAX_PWR_LVL2
				Max Power setting allowed for ConfigTDP Level 2.
		62:47		PKG_MIN_PWR_LVL2
				MIN Power setting allowed for ConfigTDP Level 2.
		63		Reserved
64BH	1611	MSR_CONFIG_TDP_CONTROL	Package	ConfigTDP Control (R/W)
		1:0		TDP_LEVEL (RW/L)
				System BIOS can program this field.
		30:2		Reserved
		31		Config_TDP_Lock (RW/L)
				When this bit is set, the content of this register is locked until a reset.
		63:32		Reserved
64CH	1612	MSR_TURBO_ACTIVATION_RATIO	Package	ConfigTDP Control (R/W)
		7:0		MAX_NON_TURBO_RATIO (RW/L)
				System BIOS can program this field.
		30:8		Reserved
		31		TURBO_ACTIVATION_RATIO_Lock (RW/L)
				When this bit is set, the content of this register is locked until a reset.
		63:32		Reserved
C80H	3200	IA32_DEBUG_INTERFACE	Package	Silicon Debug Feature Control (R/W)
				See Table 2-2.

2.13.1 MSRs in 4th Generation Intel[®] Core[™] Processor Family (based on Haswell Microarchitecture)

Table 2-30 lists model-specific registers (MSRs) that are specific to 4th generation Intel[®] Core[™] processor family and Intel[®] Xeon[®] processor E3-1200 v3 product family (based on Haswell microarchitecture). These processors have a CPUID signature with DisplayFamily_DisplayModel of 06_3 CH/ 06_4 5H/ 06_4 6H, see Table 2-1.

Table 2-30. MSRs Supported by 4th Generation Intel® Core™ Processors (Haswell Microarchitecture)

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
E2H	226	MSR_PKG_CST_CONFIG_CONTROL	Соге	C-State Configuration Control (R/W)
				Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-states.
				See http://biosbits.org.
		3:0		Package C-State Limit (R/W)
				Specifies the lowest processor-specific C-state code name (consuming the least power) for the package. The default is set as factory-configured package C-state limit.
				The following C-state code name encodings are supported:
				0000b: CO/C1 (no package C-state support)
				0001b: C2
				0010b: C3
				0011b: C6
				0100b: C7
				0101b: C7s
				Package C states C7 are not available to processors with a signature of 06_3CH.
		9:4		Reserved
		10		I/O MWAIT Redirection Enable (R/W)
		14:11		Reserved
		15		CFG Lock (R/WO)
		24:16		Reserved
		25		C3 State Auto Demotion Enable (R/W)
		26		C1 State Auto Demotion Enable (R/W)
		27		Enable C3 Undemotion (R/W)
		28		Enable C1 Undemotion (R/W)
		63:29		Reserved
17DH	381	MSR_SMM_MCA_CAP	THREAD	Enhanced SMM Capabilities (SMM-RO)
				Reports SMM capability Enhancement. Accessible only while in SMM.
		57:0		Reserved

Table 2-30. MSRs Supported by 4th Generation Intel® Core™ Processors (Haswell Microarchitecture) (Contd.)

Regi Add	ister	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		58		SMM_Code_Access_Chk (SMM-RO)
				If set to 1, indicates that the SMM code access restriction is supported and the MSR_SMM_FEATURE_CONTROL is supported.
		59		Long_Flow_Indication (SMM-RO)
				If set to 1, indicates that the SMM long flow indicator is supported and the MSR_SMM_DELAYED is supported.
		63:60		Reserved
1ADH	429	MSR_TURBO_RATIO_LIMIT	Package	Maximum Ratio Limit of Turbo Mode R/O if MSR_PLATFORM_INFO.[28] = 0.
				R/W if MSR_PLATFORM_INFO.[28] = 1.
		7:0	Package	Maximum Ratio Limit for 1C Maximum turbo ratio limit of 1 core active.
		15:8	Package	Maximum Ratio Limit for 2C
				Maximum turbo ratio limit of 2 core active.
		23:16	Package	Maximum Ratio Limit for 3C
				Maximum turbo ratio limit of 3 core active.
		31:24	Package	Maximum Ratio Limit for 4C
				Maximum turbo ratio limit of 4 core active.
		63:32		Reserved
391H	913	MSR_UNC_PERF_GLOBAL_CTRL	Package	Uncore PMU Global Control
		0		Core 0 select.
		1		Core 1 select.
		2		Core 2 select.
		3		Core 3 select.
		18:4		Reserved
		29		Enable all uncore counters.
		30		Enable wake on PMI.
		31		Enable Freezing counter when overflow.
		63:32		Reserved
392H	914	MSR_UNC_PERF_GLOBAL_STATUS	Package	Uncore PMU Main Status
		0		Fixed counter overflowed.
		1		An ARB counter overflowed.
		2		Reserved
		3		A CBox counter overflowed (on any slice).
		63:4		Reserved
394H	916	MSR_UNC_PERF_FIXED_CTRL	Package	Uncore Fixed Counter Control (R/W)
		19:0		Reserved

Table 2-30. MSRs Supported by 4th Generation Intel® Core™ Processors (Haswell Microarchitecture) (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	1		
		20		Enable overflow propagation.
		21		Reserved
		22		Enable counting.
		63:23		Reserved
395H	917	MSR_UNC_PERF_FIXED_CTR	Package	Uncore Fixed Counter
		47:0		Current count.
		63:48		Reserved
396H	918	MSR_UNC_CBO_CONFIG	Package	Uncore C-Box Configuration Information (R/O)
		3:0		Encoded number of C-Box, derive value by "-1".
		63:4		Reserved
3B0H	946	MSR_UNC_ARB_PERFCTR0	Package	Uncore Arb Unit, Performance Counter 0
3B1H	947	MSR_UNC_ARB_PERFCTR1	Package	Uncore Arb Unit, Performance Counter 1
ЗВ2Н	944	MSR_UNC_ARB_PERFEVTSEL0	Package	Uncore Arb Unit, Counter O Event Select MSR
звзн	945	MSR_UNC_ARB_PERFEVTSEL1	Package	Uncore Arb Unit, Counter 1 Event Select MSR
391H	913	MSR_UNC_PERF_GLOBAL_CTRL	Package	Uncore PMU Global Control
		0		Core 0 select.
		1		Core 1 select.
		2		Core 2 select.
		3		Core 3 select.
		18:4		Reserved
		29		Enable all uncore counters.
		30		Enable wake on PMI.
		31		Enable Freezing counter when overflow.
		63:32		Reserved
395H	917	MSR_UNC_PERF_FIXED_CTR	Package	Uncore Fixed Counter
		47:0		Current count.
		63:48		Reserved
3B3H	945	MSR_UNC_ARB_PERFEVTSEL1	Package	Uncore Arb Unit, Counter 1 Event Select MSR
4E0H	1248	MSR_SMM_FEATURE_CONTROL	Package	Enhanced SMM Feature Control (SMM-RW)
				Reports SMM capability Enhancement. Accessible only while in SMM.
		0		Lock (SMM-RWO)
				When set to '1' locks this register from further changes.
		1		Reserved

Table 2-30. MSRs Supported by 4th Generation Intel® Core™ Processors (Haswell Microarchitecture) (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		2		SMM_Code_Chk_En (SMM-RW) This control bit is available only if MSR_SMM_MCA_CAP[58] == 1. When set to '0' (default) none of the logical processors are prevented from executing SMM code outside the ranges defined by the SMRR.
				When set to '1' any logical processor in the package that attempts to execute SMM code not within the ranges defined by the SMRR will assert an unrecoverable MCE.
		63:3		Reserved
4E2H	1250	MSR_SMM_DELAYED	Package	SMM Delayed (SMM-RO)
				Reports the interruptible state of all logical processors in the package. Available only while in SMM and MSR_SMM_MCA_CAP[LONG_FLOW_INDICATION] == 1.
		N-1:0		LOG_PROC_STATE (SMM-RO)
				Each bit represents a logical processor of its state in a long flow of internal operation which delays servicing an interrupt. The corresponding bit will be set at the start of long events such as: Microcode Update Load, C6, WBINVD, Ratio Change, Throttle.
				The bit is automatically cleared at the end of each long event. The reset value of this field is 0.
				Only bit positions below N = CPUID.(EAX=0BH, ECX=PKG_LVL):EBX[15:0] can be updated.
		63:N		Reserved
4E3H	1251	MSR_SMM_BLOCKED	Package	SMM Blocked (SMM-RO)
				Reports the blocked state of all logical processors in the package. Available only while in SMM.
		N-1:0		LOG_PROC_STATE (SMM-RO)
				Each bit represents a logical processor of its blocked state to service an SMI. The corresponding bit will be set if the logical processor is in one of the following states: Wait For SIPI or SENTER Sleep.
				The reset value of this field is OFFFH.
				Only bit positions below N = CPUID.(EAX=0BH, ECX=PKG_LVL):EBX[15:0] can be updated.
		63:N		Reserved
606H	1542	MSR_RAPL_POWER_UNIT	Package	Unit Multipliers Used in RAPL Interfaces (R/O)
		3:0	Package	Power Units
				See Section 15.10.1, "RAPL Interfaces."
		7:4	Package	Reserved

Table 2-30. MSRs Supported by 4th Generation Intel® Core™ Processors (Haswell Microarchitecture) (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	1		
		12:8	Package	Energy Status Units Energy related information (in Joules) is based on the multiplier, 1/2^ESU; where ESU is an unsigned integer represented by bits 12:8. Default value is OEH (or 61 micro-joules).
		15:13	Package	Reserved
		19:16	Package	Time Units See Section 15.10.1, "RAPL Interfaces."
		63:20		Reserved
639H	1593	MSR_PPO_ENERGY_STATUS	Package	PPO Energy Status (R/O) See Section 15.10.4, "PPO/PP1 RAPL Domains."
640H	1600	MSR_PP1_POWER_LIMIT	Package	PP1 RAPL Power Limit Control (R/W) See Section 15.10.4, "PP0/PP1 RAPL Domains."
641H	1601	MSR_PP1_ENERGY_STATUS	Package	PP1 Energy Status (R/O) See Section 15.10.4, "PP0/PP1 RAPL Domains."
642H	1602	MSR_PP1_POLICY	Package	PP1 Balance Policy (R/W) See Section 15.10.4, "PP0/PP1 RAPL Domains."
690H	1680	MSR_CORE_PERF_LIMIT_REASONS	Package	Indicator of Frequency Clipping in Processor Cores (R/W)
				(Frequency refers to processor core frequency.)
		0		PROCHOT Status (R0) When set, processor core frequency is reduced below the operating system request due to assertion of external PROCHOT.
		1		Thermal Status (R0)
				When set, frequency is reduced below the operating system request due to a thermal event.
		3:2		Reserved
		4		Graphics Driver Status (R0)
				When set, frequency is reduced below the operating system request due to Processor Graphics driver override.
		5		Autonomous Utilization-Based Frequency Control Status (RO)
				When set, frequency is reduced below the operating system request because the processor has detected that utilization is low.
		6		VR Therm Alert Status (R0)
				When set, frequency is reduced below the operating system request due to a thermal alert from the Voltage Regulator.
		7		Reserved

Table 2-30. MSRs Supported by 4th Generation Intel® Core™ Processors (Haswell Microarchitecture) (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		8		Electrical Design Point Status (R0) When set, frequency is reduced below the operating system request due to electrical design point constraints (e.g., maximum electrical current consumption).
		9		Core Power Limiting Status (R0) When set, frequency is reduced below the operating system request due to domain-level power limiting.
		10		Package-Level Power Limiting PL1 Status (R0)
				When set, frequency is reduced below the operating system request due to package-level power limiting PL1.
		11		Package-Level PL2 Power Limiting Status (R0)
				When set, frequency is reduced below the operating system request due to package-level power limiting PL2.
		12		Max Turbo Limit Status (RO)
				When set, frequency is reduced below the operating system request due to multi-core turbo limits.
		13		Turbo Transition Attenuation Status (R0)
				When set, frequency is reduced below the operating system request due to Turbo transition attenuation. This prevents performance degradation due to frequent operating ratio changes.
		15:14		Reserved
		16		PROCHOT Log
				When set, indicates that the PROCHOT Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		17		Thermal Log
				When set, indicates that the Thermal Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		19:18		Reserved
		20		Graphics Driver Log
				When set, indicates that the Graphics Driver Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.

Table 2-30. MSRs Supported by 4th Generation Intel® Core™ Processors (Haswell Microarchitecture) (Contd.)

Regi Addı	ister	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		21		Autonomous Utilization-Based Frequency Control Log
				When set, indicates that the Autonomous Utilization- Based Frequency Control Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		22		VR Therm Alert Log
				When set, indicates that the VR Therm Alert Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		23		Reserved
		24		Electrical Design Point Log
				When set, indicates that the EDP Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		25		Core Power Limiting Log
				When set, indicates that the Core Power Limiting Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		26		Package-Level PL1 Power Limiting Log
				When set, indicates that the Package Level PL1 Power Limiting Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		27		Package-Level PL2 Power Limiting Log
				When set, indicates that the Package Level PL2 Power Limiting Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		28		Max Turbo Limit Log
				When set, indicates that the Max Turbo Limit Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.

Table 2-30. MSRs Supported by 4th Generation Intel® Core™ Processors (Haswell Microarchitecture) (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		29		Turbo Transition Attenuation Log When set, indicates that the Turbo Transition Attenuation Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		63:30		Reserved
6B0H	1712	MSR_GRAPHICS_PERF_LIMIT_REASONS	Package	Indicator of Frequency Clipping in the Processor Graphics (R/W)
				(Frequency refers to processor graphics frequency.)
		0		PROCHOT Status (R0)
				When set, frequency is reduced below the operating system request due to assertion of external PROCHOT.
		1		Thermal Status (R0)
				When set, frequency is reduced below the operating system request due to a thermal event.
		3:2		Reserved
		4		Graphics Driver Status (R0)
				When set, frequency is reduced below the operating system request due to Processor Graphics driver override.
		5		Autonomous Utilization-Based Frequency Control Status (RO)
				When set, frequency is reduced below the operating system request because the processor has detected that utilization is low.
		6		VR Therm Alert Status (R0)
				When set, frequency is reduced below the operating system request due to a thermal alert from the Voltage Regulator.
		7		Reserved
		8		Electrical Design Point Status (R0)
				When set, frequency is reduced below the operating system request due to electrical design point constraints (e.g., maximum electrical current consumption).
		9		Graphics Power Limiting Status (R0)
				When set, frequency is reduced below the operating system request due to domain-level power limiting.
		10		Package-Level Power Limiting PL1 Status (R0)
				When set, frequency is reduced below the operating system request due to package-level power limiting PL1.

Table 2-30. MSRs Supported by 4th Generation Intel® Core™ Processors (Haswell Microarchitecture) (Contd.)

Regi Addı	ster	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	1		
		11		Package-Level PL2 Power Limiting Status (R0) When set, frequency is reduced below the operating system request due to package-level power limiting PL2.
		15:12		Reserved
		16		PROCHOT Log
				When set, indicates that the PROCHOT Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		17		Thermal Log
				When set, indicates that the Thermal Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		19:18		Reserved
		20		Graphics Driver Log
				When set, indicates that the Graphics Driver Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		21		Autonomous Utilization-Based Frequency Control Log
				When set, indicates that the Autonomous Utilization- Based Frequency Control Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		22		VR Therm Alert Log
				When set, indicates that the VR Therm Alert Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		23		Reserved
		24		Electrical Design Point Log
				When set, indicates that the EDP Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		25		Core Power Limiting Log
				When set, indicates that the Core Power Limiting Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.

Table 2-30. MSRs Supported by 4th Generation Intel® Core™ Processors (Haswell Microarchitecture) (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		26		Package-Level PL1 Power Limiting Log When set, indicates that the Package Level PL1 Power Limiting Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software
		27		writing 0.
		27		Package-Level PL2 Power Limiting Log When set, indicates that the Package Level PL2 Power Limiting Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		28		Max Turbo Limit Log
				When set, indicates that the Max Turbo Limit Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		29		Turbo Transition Attenuation Log
				When set, indicates that the Turbo Transition Attenuation Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		63:30		Reserved
6B1H	1713	MSR_RING_PERF_LIMIT_REASONS	Package	Indicator of Frequency Clipping in the Ring Interconnect (R/W)
				(Frequency refers to ring interconnect in the uncore.)
		0		PROCHOT Status (R0)
				When set, frequency is reduced below the operating system request due to assertion of external PROCHOT.
		1		Thermal Status (R0)
				When set, frequency is reduced below the operating system request due to a thermal event.
		5:2		Reserved
		6		VR Therm Alert Status (R0)
				When set, frequency is reduced below the operating system request due to a thermal alert from the Voltage Regulator.
		7		Reserved
		8		Electrical Design Point Status (R0)
				When set, frequency is reduced below the operating system request due to electrical design point constraints (e.g., maximum electrical current consumption).

Table 2-30. MSRs Supported by 4th Generation Intel® Core™ Processors (Haswell Microarchitecture) (Contd.)

Regi Add	ister	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	7		
		9		Reserved
		10		Package-Level Power Limiting PL1 Status (R0)
				When set, frequency is reduced below the operating system request due to package-level power limiting PL1.
		11		Package-Level PL2 Power Limiting Status (R0)
				When set, frequency is reduced below the operating system request due to package-level power limiting PL2.
		15:12		Reserved
		16		PROCHOT Log
				When set, indicates that the PROCHOT Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		17		Thermal Log
				When set, indicates that the Thermal Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		19:18		Reserved.
		20		Graphics Driver Log
				When set, indicates that the Graphics Driver Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		21		Autonomous Utilization-Based Frequency Control Log
				When set, indicates that the Autonomous Utilization- Based Frequency Control Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		22		VR Therm Alert Log
				When set, indicates that the VR Therm Alert Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		23		Reserved
		24		Electrical Design Point Log
				When set, indicates that the EDP Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.

Table 2-30. MSRs Supported by 4th Generation Intel® Core™ Processors (Haswell Microarchitecture) (Contd.)

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		25		Core Power Limiting Log When set, indicates that the Core Power Limiting Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.
		26		Package-Level PL1 Power Limiting Log
				When set, indicates that the Package Level PL1 Power Limiting Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		27		Package-Level PL2 Power Limiting Log
				When set, indicates that the Package Level PL2 Power Limiting Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		28		Max Turbo Limit Log
				When set, indicates that the Max Turbo Limit Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software
				writing 0.
		29		Turbo Transition Attenuation Log When set, indicates that the Turbo Transition Attenuation Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		63:30		Reserved
700H	1792	MSR_UNC_CBO_0_PERFEVTSEL0	Package	Uncore C-Box 0, Counter 0 Event Select MSR
701H	1793	MSR_UNC_CBO_0_PERFEVTSEL1	Package	Uncore C-Box 0, Counter 1 Event Select MSR
706H	1798	MSR_UNC_CBO_0_PERFCTR0	Package	Uncore C-Box 0, Performance Counter 0
707H	1799	MSR_UNC_CBO_0_PERFCTR1	Package	Uncore C-Box 0, Performance Counter 1
710H	1808	MSR_UNC_CBO_1_PERFEVTSELO	Package	Uncore C-Box 1, Counter 0 Event Select MSR
711H	1809	MSR_UNC_CBO_1_PERFEVTSEL1	Package	Uncore C-Box 1, Counter 1 Event Select MSR
716H	1814	MSR_UNC_CBO_1_PERFCTR0	Package	Uncore C-Box 1, Performance Counter 0
717H	1815	MSR_UNC_CBO_1_PERFCTR1	Package	Uncore C-Box 1, Performance Counter 1
720H	1824	MSR_UNC_CBO_2_PERFEVTSEL0	Package	Uncore C-Box 2, Counter 0 Event Select MSR
721H	1824	MSR_UNC_CBO_2_PERFEVTSEL1	Package	Uncore C-Box 2, Counter 1 Event Select MSR
726H	1830	MSR_UNC_CBO_2_PERFCTR0	Package	Uncore C-Box 2, Performance Counter 0
727H	1831	MSR_UNC_CBO_2_PERFCTR1	Package	Uncore C-Box 2, Performance Counter 1

Table 2-30. MSRs Supported by 4th Generation Intel® Core™ Processors (Haswell Microarchitecture) (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
730H	1840	MSR_UNC_CBO_3_PERFEVTSELO	Package	Uncore C-Box 3, Counter 0 Event Select MSR
731H	1841	MSR_UNC_CBO_3_PERFEVTSEL1	Package	Uncore C-Box 3, Counter 1 Event Select MSR
736H	1846	MSR_UNC_CBO_3_PERFCTRO	Package	Uncore C-Box 3, Performance Counter 0
737H	1847	MSR_UNC_CBO_3_PERFCTR1	Package	Uncore C-Box 3, Performance Counter 1

See Table 2-20, Table 2-21, Table 2-22, Table 2-25, Table 2-29 for other MSR definitions applicable to processors with CPUID signatures 063CH, 06_46H.

2.13.2 Additional Residency MSRs Supported in 4th Generation Intel® Core™ Processors

The 4th generation Intel[®] Core[™] processor family (based on Haswell microarchitecture) with CPUID DisplayFamily_DisplayModel signature 06_45H supports the MSR interfaces listed in Table 2-20, Table 2-21, Table 2-29, Table 2-30, and Table 2-31.

Table 2-31. Additional Residency MSRs Supported by 4th Generation Intel® Core™ Processors with DisplayFamily_DisplayModel Signature 06_45H

Regi Addı		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
E2H	226	MSR_PKG_CST_CONFIG_CONTROL	Соге	C-State Configuration Control (R/W)
				Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-states.
				See http://biosbits.org.
		3:0		Package C-State Limit (R/W)
				Specifies the lowest processor-specific C-state code name (consuming the least power) for the package. The default is set as factory-configured package C-state limit.
				The following C-state code name encodings are supported:
				0000b: C0/C1 (no package C-state support)
				0001b: C2
				0010b: C3
				0011b: C6
				0100b: C7
				0101b: C7s
				0110b: C8
				0111b: C9
				1000b: C10
		9:4		Reserved
		10		I/O MWAIT Redirection Enable (R/W)
		14:11		Reserved
		15		CFG Lock (R/W0)

Table 2-31. Additional Residency MSRs Supported by 4th Generation Intel® Core™ Processors with DisplayFamily_DisplayModel Signature 06_45H

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		24:16		Reserved
		25		C3 State Auto Demotion Enable (R/W)
		26		C1 State Auto Demotion Enable (R/W)
		27		Enable C3 Undemotion (R/W)
		28		Enable C1 Undemotion (R/W)
		63:29		Reserved
630H	1584	MSR_PKG_C8_RESIDENCY	Package	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		59:0		Package C8 Residency Counter (R/O)
				Value since last reset that this package is in processor- specific C8 states. Count at the same frequency as the TSC.
		63:60		Reserved
631H	1585	MSR_PKG_C9_RESIDENCY	Package	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		59:0		Package C9 Residency Counter (R/O)
				Value since last reset that this package is in processor- specific C9 states. Count at the same frequency as the TSC.
		63:60		Reserved
632H	1586	MSR_PKG_C10_RESIDENCY	Package	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		59:0		Package C10 Residency Counter (R/O)
				Value since last reset that this package is in processor-specific C10 states. Count at the same frequency as the TSC.
		63:60		Reserved

See Table 2-20, Table 2-21, Table 2-22, Table 2-29, Table 2-30 for other MSR definitions applicable to processors with CPUID signature 06_45H.

2.14 MSRS IN INTEL® XEON® PROCESSOR E5 V3 AND E7 V3 PRODUCT FAMILY

Intel[®] Xeon[®] processor E5 v3 family and Intel[®] Xeon[®] processor E7 v3 family are based on Haswell-E microarchitecture (CPUID DisplayFamily_DisplayModel = 06_3 F). These processors supports the MSR interfaces listed in Table 2-20, Table 2-29, and Table 2-32.

Table 2-32. Additional MSRs Supported by Intel® Xeon® Processor E5 v3 Family

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
35H	53	MSR_CORE_THREAD_COUNT	Package	Configured State of Enabled Processor Core Count and Logical Processor Count (R/O)
				 After a Power-On RESET, enumerates factory configuration of the number of processor cores and logical processors in the physical package. Following the sequence of (i) BIOS modified a Configuration Mask which selects a subset of processor cores to be active post RESET and (ii) a RESET event after the modification, enumerates the current configuration of enabled processor core count and logical processor count in the physical package.
		15:0		THREAD_COUNT (R/O)
				The number of logical processors that are currently enabled (by either factory configuration or BIOS configuration) in the physical package.
		31:16		Core_COUNT (R/O)
				The number of processor cores that are currently enabled (by either factory configuration or BIOS configuration) in the physical package.
		63:32		Reserved
53H	83	MSR_THREAD_ID_INFO	Thread	A Hardware Assigned ID for the Logical Processor (R/O)
		7:0		Logical_Processor_ID (R/O)
				An implementation-specific numerical value physically assigned to each logical processor. This ID is not related to Initial APIC ID or x2APIC ID, it is unique within a physical package.
		63:8		Reserved
E2H	226	MSR_PKG_CST_CONFIG_CONTROL	Соге	C-State Configuration Control (R/W)
				Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-states.
				See http://biosbits.org.
		2:0		Package C-State Limit (R/W)
				Specifies the lowest processor-specific C-state code name (consuming the least power) for the package. The default is set as factory-configured package C-state limit.
				The following C-state code name encodings are supported:
				000b: CO/C1 (no package C-state support)
				001b: C2
				010b: C6 (non-retention)
				011b: C6 (retention)
				111b: No Package C state limits. All C states supported by the processor are available.

Table 2-32. Additional MSRs Supported by Intel® Xeon® Processor E5 v3 Family

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		9:3		Reserved
		10		I/O MWAIT Redirection Enable (R/W)
		14:11		Reserved
		15		CFG Lock (R/W0)
		24:16		Reserved
		25		C3 State Auto Demotion Enable (R/W)
		26		C1 State Auto Demotion Enable (R/W)
		27		Enable C3 Undemotion (R/W)
		28		Enable C1 Undemotion (R/W)
		29		Package C State Demotion Enable (R/W)
		30		Package C State UnDemotion Enable (R/W)
		63:31		Reserved
179H	377	IA32_MCG_CAP	Thread	Global Machine Check Capability (R/O)
		7:0		Count
		8		MCG_CTL_P
		9		MCG_EXT_P
		10		MCP_CMCI_P
		11		MCG_TES_P
		15:12		Reserved
		23:16		MCG_EXT_CNT
		24		MCG_SER_P
		25		MCG_EM_P
		26		MCG_ELOG_P
		63:27		Reserved
17DH	381	MSR_SMM_MCA_CAP	Thread	Enhanced SMM Capabilities (SMM-RO)
				Reports SMM capability Enhancement. Accessible only while in SMM.
		57:0		Reserved
		58		SMM_Code_Access_Chk (SMM-RO)
				If set to 1, indicates that the SMM code access restriction is supported and a host-space interface available to SMM handler.
		59		Long_Flow_Indication (SMM-RO)
				If set to 1, indicates that the SMM long flow indicator is supported and a host-space interface available to SMM handler.
		63:60		Reserved
17FH	383	MSR_ERROR_CONTROL	Package	MC Bank Error Configuration (R/W)
		0		Reserved

Table 2-32. Additional MSRs Supported by Intel® Xeon® Processor E5 v3 Family

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		1		MemError Log Enable (R/W) When set, enables IMC status bank to log additional info in bits 36:32.
		63:2		Reserved
1ADH	429	MSR_TURBO_RATIO_LIMIT	Package	Maximum Ratio Limit of Turbo Mode R/O if MSR_PLATFORM_INFO.[28] = 0. R/W if MSR_PLATFORM_INFO.[28] = 1.
		7:0	Package	Maximum Ratio Limit for 1C Maximum turbo ratio limit of 1 core active.
		15:8	Package	Maximum Ratio Limit for 2C Maximum turbo ratio limit of 2 core active.
		23:16	Package	Maximum Ratio Limit for 3C Maximum turbo ratio limit of 3 core active.
		31:24	Package	Maximum Ratio Limit for 4C Maximum turbo ratio limit of 4 core active.
		39:32	Package	Maximum Ratio Limit for 5C Maximum turbo ratio limit of 5 core active.
		47:40	Package	Maximum Ratio Limit for 6C Maximum turbo ratio limit of 6 core active.
		55:48	Package	Maximum Ratio Limit for 7C Maximum turbo ratio limit of 7 core active.
		63:56	Package	Maximum Ratio Limit for 8C Maximum turbo ratio limit of 8 core active.
1AEH	430	MSR_TURBO_RATIO_LIMIT1	Package	Maximum Ratio Limit of Turbo Mode R/O if MSR_PLATFORM_INFO.[28] = 0. R/W if MSR_PLATFORM_INFO.[28] = 1.
		7:0	Package	Maximum Ratio Limit for 9C Maximum turbo ratio limit of 9 core active.
		15:8	Package	Maximum Ratio Limit for 10C Maximum turbo ratio limit of 10 core active.
		23:16	Package	Maximum Ratio Limit for 11C Maximum turbo ratio limit of 11 core active.
		31:24	Package	Maximum Ratio Limit for 12C Maximum turbo ratio limit of 12 core active.
		39:32	Package	Maximum Ratio Limit for 13C Maximum turbo ratio limit of 13 core active.
		47:40	Package	Maximum Ratio Limit for 14C Maximum turbo ratio limit of 14 core active.
		55:48	Package	Maximum Ratio Limit for 15C Maximum turbo ratio limit of 15 core active.

Table 2-32. Additional MSRs Supported by Intel® Xeon® Processor E5 v3 Family

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		63:56	Package	Maximum Ratio Limit for 16C Maximum turbo ratio limit of 16 core active.
1AFH	431	431 MSR_TURBO_RATIO_LIMIT2	Package	Maximum Ratio Limit of Turbo Mode R/O if MSR_PLATFORM_INFO.[28] = 0. R/W if MSR_PLATFORM_INFO.[28] = 1.
		7:0	Package	Maximum Ratio Limit for 17C Maximum turbo ratio limit of 17 core active.
		15:8	Package	Maximum Ratio Limit for 18C Maximum turbo ratio limit of 18 core active.
		62:16	Package	Reserved
		63	Package	Semaphore for Turbo Ratio Limit Configuration If 1, the processor uses override configuration specified in MSR_TURBO_RATIO_LIMIT, MSR_TURBO_RATIO_LIMIT1, and MSR_TURBO_RATIO_LIMIT2.
				If 0, the processor uses factory-set configuration (Default).
414H	1044	IA32_MC5_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
415H	1045	IA32_MC5_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.". Bank MC5 reports MC errors from the Intel QPI 0
416H	1046	IA32_MC5_ADDR	Package	module.
417H	1047	IA32_MC5_MISC	Package	
418H	1048	IA32_MC6_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
419H	1049	IA32_MC6_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.". Bank MC6 reports MC errors from the integrated I/O
41AH	1050	IA32_MC6_ADDR	Package	module.
41BH	1051	IA32_MC6_MISC	Package	
41CH	1052	IA32_MC7_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
41DH	1053	IA32_MC7_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.". Bank MC7 reports MC errors from the home agent HA 0.
41EH	1054	IA32_MC7_ADDR	Package	Bank 1107 reports 110 errors from the frome agent 11/10.
41FH	1055	IA32_MC7_MISC	Package	
420H	1056	IA32_MC8_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
421H	1057	IA32_MC8_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.". Bank MC8 reports MC errors from the home agent HA 1.
422H	1058	IA32_MC8_ADDR	Package	Bank i ico reports i ic errors from the nome agent inv i.
423H	1059	IA32_MC8_MISC	Package	
424H	1060	IA32_MC9_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
425H	1061	IA32_MC9_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.". Banks MC9 through MC 16 report MC errors from each
426H	1062	IA32_MC9_ADDR	Package	channel of the integrated memory controllers.
427H	1063	IA32_MC9_MISC	Package	

Table 2-32. Additional MSRs Supported by Intel® Xeon® Processor E5 v3 Family

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
428H	1064	IA32_MC10_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
429H	1065	IA32_MC10_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
42AH	1066	IA32_MC10_ADDR	Package	Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.
42BH	1067	IA32_MC10_MISC	Package	
42CH	1068	IA32_MC11_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
42DH	1069	IA32_MC11_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
42EH	1070	IA32_MC11_ADDR	Package	Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.
42FH	1071	IA32_MC11_MISC	Package	
430H	1072	IA32_MC12_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
431H	1073	IA32_MC12_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
432H	1074	IA32_MC12_ADDR	Package	Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.
433H	1075	IA32_MC12_MISC	Package	
434H	1076	IA32_MC13_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
435H	1077	IA32_MC13_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
436H	1078	IA32_MC13_ADDR	Package	Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.
437H	1079	IA32_MC13_MISC	Package	
438H	1080	IA32_MC14_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
439H	1081	IA32_MC14_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
43AH	1082	IA32_MC14_ADDR	Package	Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.
43BH	1083	IA32_MC14_MISC	Package	
43CH	1084	IA32_MC15_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
43DH	1085	IA32_MC15_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
43EH	1086	IA32_MC15_ADDR	Package	Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.
43FH	1087	IA32_MC15_MISC	Package	
440H	1088	IA32_MC16_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
441H	1089	IA32_MC16_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
442H	1090	IA32_MC16_ADDR	Package	Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.
443H	1091	IA32_MC16_MISC	Package	
444H	1092	IA32_MC17_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
445H	1093	IA32_MC17_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
446H	1094	IA32_MC17_ADDR	Package	Bank MC17 reports MC errors from the following pair of CBo/L3 Slices (if the pair is present): CBo0, CBo3, CBo6,
447H	1095	IA32_MC17_MISC	Package	CBo9, CBo12, CBo15.
448H	1096	IA32_MC18_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
449H	1097	IA32_MC18_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
44AH	1098	IA32_MC18_ADDR	Package	Bank MC18 reports MC errors from the following pair of CBo/L3 Slices (if the pair is present): CBo1, CBo4, CBo7,
44BH	1099	IA32_MC18_MISC	Package	CBo10, CBo13, CBo16.

Table 2-32. Additional MSRs Supported by Intel® Xeon® Processor E5 v3 Family

Regi Add	ister ress	Register Name / Bit Fields	Scope	Bit Description	
Hex	Dec				
44CH	1100	IA32_MC19_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through	
44DH	1101	IA32_MC19_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.". Bank MC19 reports MC errors from the following pair of CBo/L3 Slices (if the pair is present): CBo2, CBo5, CBo8	
44EH	1102	IA32_MC19_ADDR	Package		
44FH	1103	IA32_MC19_MISC	Package	CBo11, CBo14, CBo17.	
450H	1104	IA32_MC20_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through	
451H	1105	IA32_MC20_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".	
452H	1106	IA32_MC20_ADDR	Package	Bank MC20 reports MC errors from the Intel QPI 1 module.	
453H	1107	IA32_MC20_MISC	Package		
454H	1108	IA32_MC21_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through	
455H	1109	IA32_MC21_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".	
456H	1110	IA32_MC21_ADDR	Package	Bank MC21 reports MC errors from the Intel QPI 2 module.	
457H	1111	IA32_MC21_MISC	Package		
606H	1542	MSR_RAPL_POWER_UNIT	Package	Unit Multipliers Used in RAPL Interfaces (R/0)	
		3:0	Package	Power Units	
				See Section 15.10.1, "RAPL Interfaces."	
		7:4	Package	Reserved	
		12:8	Package	Energy Status Units	
					Energy related information (in Joules) is based on the multiplier, 1/2^ESU; where ESU is an unsigned integer represented by bits 12:8. Default value is 0EH (or 61 micro-joules).
		15:13	Package	Reserved	
		19:16	Package	Time Units	
				See Section 15.10.1, "RAPL Interfaces."	
		63:20		Reserved	
618H	1560	MSR_DRAM_POWER_LIMIT	Package	DRAM RAPL Power Limit Control (R/W)	
				See Section 15.10.5, "DRAM RAPL Domain."	
619H	1561	MSR_DRAM_ENERGY_STATUS	Package	DRAM Energy Status (R/O)	
				Energy Consumed by DRAM devices.	
		31:0		Energy in 15.3 micro-joules. Requires BIOS configuration to enable DRAM RAPL mode 0 (Direct VR).	
		63:32		Reserved	
61BH	1563	MSR_DRAM_PERF_STATUS	Package	DRAM Performance Throttling Status (R/0)	
				See Section 15.10.5, "DRAM RAPL Domain."	
61CH	1564	MSR_DRAM_POWER_INFO	Package	DRAM RAPL Parameters (R/W)	
				See Section 15.10.5, "DRAM RAPL Domain."	
61EH	1566	MSR_PCIE_PLL_RATIO	Package	Configuration of PCIE PLL Relative to BCLK(R/W)	

Table 2-32. Additional MSRs Supported by Intel® Xeon® Processor E5 v3 Family

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		1:0	Package	PCIE Ratio (R/W) 00b: Use 5:5 mapping for100MHz operation (default). 01b: Use 5:4 mapping for125MHz operation. 10b: Use 5:3 mapping for166MHz operation. 11b: Use 5:2 mapping for250MHz operation.
		2	Package	LPLL Select (R/W) If 1, use configured setting of PCIE Ratio.
		3	Package	LONG RESET (R/W) If 1, wait an additional time-out before re-locking Gen2/Gen3 PLLs.
		63:4		Reserved
620H	1568	MSR UNCORE_RATIO_LIMIT	Package	Uncore Ratio Limit (R/W) Out of reset, the min_ratio and max_ratio fields represent the widest possible range of uncore frequencies. Writing to these fields allows software to control the minimum and the maximum frequency that hardware will select.
		63:15		Reserved
		14:8		MIN_RATIO Writing to this field controls the minimum possible ratio of the LLC/Ring.
		7		Reserved
		6:0		MAX_RATIO This field is used to limit the max ratio of the LLC/Ring.
639H	1593	MSR_PPO_ENERGY_STATUS	Package	Reserved (R/O) Reads return 0.
690H	1680	MSR_CORE_PERF_LIMIT_REASONS	Package	Indicator of Frequency Clipping in Processor Cores (R/W) (Frequency refers to processor core frequency.)
		0		PROCHOT Status (R0) When set, processor core frequency is reduced below the operating system request due to assertion of external PROCHOT.
		1		Thermal Status (R0) When set, frequency is reduced below the operating system request due to a thermal event.
		2		Power Budget Management Status (R0) When set, frequency is reduced below the operating system request due to PBM limit
		3		Platform Configuration Services Status (R0) When set, frequency is reduced below the operating system request due to PCS limit
		4		Reserved

Table 2-32. Additional MSRs Supported by Intel® Xeon® Processor E5 v3 Family

Register Address		Register Name / Bit Fields	Scope	Bit Description	
Hex	Dec				
		5		Autonomous Utilization-Based Frequency Control Status (RO)	
				When set, frequency is reduced below the operating system request because the processor has detected that utilization is low.	
		6		VR Therm Alert Status (R0)	
				When set, frequency is reduced below the operating system request due to a thermal alert from the Voltage Regulator.	
		7		Reserved	
		8		Electrical Design Point Status (R0)	
				When set, frequency is reduced below the operating system request due to electrical design point constraints (e.g., maximum electrical current consumption).	
		9		Reserved	
		10		Multi-Core Turbo Status (R0)	
				When set, frequency is reduced below the operating system request due to Multi-Core Turbo limits.	
		12:11		Reserved	
		13		Core Frequency P1 Status (R0)	
				When set, frequency is reduced below max non-turbo P1.	
		14		Core Max N-Core Turbo Frequency Limiting Status (R0)	
				When set, frequency is reduced below max n-core turbo frequency.	
		15		Core Frequency Limiting Status (R0)	
				When set, frequency is reduced below the operating system request.	
		16		PROCHOT Log	
				When set, indicates that the PROCHOT Status bit has asserted since the log bit was last cleared.	
				This log bit will remain set until cleared by software writing 0.	
		17		Thermal Log	
				When set, indicates that the Thermal Status bit has asserted since the log bit was last cleared.	
				This log bit will remain set until cleared by software writing 0.	
		18		Power Budget Management Log	
				When set, indicates that the PBM Status bit has asserted since the log bit was last cleared.	
				This log bit will remain set until cleared by software writing 0.	

Table 2-32. Additional MSRs Supported by Intel® Xeon® Processor E5 v3 Family

Regi Addı		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		19		Platform Configuration Services Log When set, indicates that the PCS Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.
		20		Reserved
		21		Autonomous Utilization-Based Frequency Control Log
				When set, indicates that the AUBFC Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		22		VR Therm Alert Log
				When set, indicates that the VR Therm Alert Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		23		Reserved
		24		Electrical Design Point Log
				When set, indicates that the EDP Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		25		Reserved
		26		Multi-Core Turbo Log
				When set, indicates that the Multi-Core Turbo Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		28:27		Reserved
		29		Core Frequency P1 Log
				When set, indicates that the Core Frequency P1 Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		30		Core Max N-Core Turbo Frequency Limiting Log
				When set, indicates that the Core Max n-core Turbo Frequency Limiting Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.

Table 2-32. Additional MSRs Supported by Intel® Xeon® Processor E5 v3 Family

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		31		Core Frequency Limiting Log
				When set, indicates that the Core Frequency Limiting Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		63:32		Reserved
C8DH	3213	IA32_QM_EVTSEL	THREAD	Monitoring Event Select Register (R/W)
				If CPUID.(EAX=07H, ECX=0):EBX.RDT-M[bit 12] = 1.
		7:0		EventID (R/W)
				Event encoding:
				0x0: No monitoring.
				0x1: L3 occupancy monitoring.
				All other encoding reserved.
		31:8		Reserved
		41:32		RMID (R/W)
		63:42		Reserved
C8EH	3214	IA32_QM_CTR	THREAD	Monitoring Counter Register (R/O)
				If CPUID.(EAX=07H, ECX=0):EBX.RDT-M[bit 12] = 1.
		61:0		Resource Monitored Data
		62		Unavailable: If 1, indicates data for this RMID is not available or not monitored for this resource or RMID.
		63		Error: If 1, indicates an unsupported RMID or event type was written to IA32_PQR_QM_EVTSEL.
C8FH	3215	IA32_PQR_ASSOC	THREAD	Resource Association Register (R/W)
		9:0		RMID
		63: 10		Reserved
See Tabl	e 2-20, Ta	able 2-29 for other MSR definitions appli	cable to process	ors with CPUID signature 06_3FH.

NOTES:

2.14.1 Additional Uncore PMU MSRs in the Intel® Xeon® Processor E5 v3 Family

Intel Xeon Processor E5 v3 and E7 v3 family are based on the Haswell-E microarchitecture. The MSR-based uncore PMU interfaces are listed in Table 2-33. For complete detail of the uncore PMU, refer to Intel Xeon Processor E5 v3 Product Family Uncore Performance Monitoring Guide. These processors have a CPUID signature with DisplayFamily_DisplayModel of 06_3 FH.

^{1.} An override configuration lower than the factory-set configuration is always supported. An override configuration higher than the factory-set configuration is dependent on features specific to the processor and the platform.

Table 2-33. Uncore PMU MSRs in Intel® Xeon® Processor E5 v3 Family

	ister Iress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
700H	1792	MSR_PMON_GLOBAL_CTL	Package	Uncore Perfmon Per-Socket Global Control
701H	1793	MSR_PMON_GLOBAL_STATUS	Package	Uncore Perfmon Per-Socket Global Status
702H	1794	MSR_PMON_GLOBAL_CONFIG	Package	Uncore Perfmon Per-Socket Global Configuration
703H	1795	MSR_U_PMON_UCLK_FIXED_CTL	Package	Uncore U-Box UCLK Fixed Counter Control
704H	1796	MSR_U_PMON_UCLK_FIXED_CTR	Package	Uncore U-Box UCLK Fixed Counter
705H	1797	MSR_U_PMON_EVNTSELO	Package	Uncore U-Box Perfmon Event Select for U-Box Counter 0
706H	1798	MSR_U_PMON_EVNTSEL1	Package	Uncore U-Box Perfmon Event Select for U-Box Counter 1
708H	1800	MSR_U_PMON_BOX_STATUS	Package	Uncore U-Box Perfmon U-Box Wide Status
709H	1801	MSR_U_PMON_CTR0	Package	Uncore U-Box Perfmon Counter 0
70AH	1802	MSR_U_PMON_CTR1	Package	Uncore U-Box Perfmon Counter 1
710H	1808	MSR_PCU_PMON_BOX_CTL	Package	Uncore PCU Perfmon for PCU-Box-Wide Control
711H	1809	MSR_PCU_PMON_EVNTSEL0	Package	Uncore PCU Perfmon Event Select for PCU Counter 0
712H	1810	MSR_PCU_PMON_EVNTSEL1	Package	Uncore PCU Perfmon Event Select for PCU Counter 1
713H	1811	MSR_PCU_PMON_EVNTSEL2	Package	Uncore PCU Perfmon Event Select for PCU Counter 2
714H	1812	MSR_PCU_PMON_EVNTSEL3	Package	Uncore PCU Perfmon Event Select for PCU Counter 3
715H	1813	MSR_PCU_PMON_BOX_FILTER	Package	Uncore PCU Perfmon Box-Wide Filter
716H	1814	MSR_PCU_PMON_BOX_STATUS	Package	Uncore PCU Perfmon Box Wide Status
717H	1815	MSR_PCU_PMON_CTR0	Package	Uncore PCU Perfmon Counter 0
718H	1816	MSR_PCU_PMON_CTR1	Package	Uncore PCU Perfmon Counter 1
719H	1817	MSR_PCU_PMON_CTR2	Package	Uncore PCU Perfmon Counter 2
71AH	1818	MSR_PCU_PMON_CTR3	Package	Uncore PCU Perfmon Counter 3
720H	1824	MSR_S0_PMON_BOX_CTL	Package	Uncore SBo 0 Perfmon for SBo 0 Box-Wide Control
721H	1825	MSR_SO_PMON_EVNTSELO	Package	Uncore SBo 0 Perfmon Event Select for SBo 0 Counter 0
722H	1826	MSR_S0_PMON_EVNTSEL1	Package	Uncore SBo 0 Perfmon Event Select for SBo 0 Counter 1
723H	1827	MSR_SO_PMON_EVNTSEL2	Package	Uncore SBo 0 Perfmon Event Select for SBo 0 Counter 2
724H	1828	MSR_SO_PMON_EVNTSEL3	Package	Uncore SBo 0 Perfmon Event Select for SBo 0 Counter 3
725H	1829	MSR_SO_PMON_BOX_FILTER	Package	Uncore SBo O Perfmon Box-Wide Filter
726H	1830	MSR_SO_PMON_CTRO	Package	Uncore SBo O Perfmon Counter O
727H	1831	MSR_S0_PMON_CTR1	Package	Uncore SBo O Perfmon Counter 1
728H	1832	MSR_S0_PMON_CTR2	Package	Uncore SBo O Perfmon Counter 2
729H	1833	MSR_SO_PMON_CTR3	Package	Uncore SBo O Perfmon Counter 3
72AH	1834	MSR_S1_PMON_BOX_CTL	Package	Uncore SBo 1 Perfmon for SBo 1 Box-Wide Control

Table 2-33. Uncore PMU MSRs in Intel® Xeon® Processor E5 v3 Family (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
72BH	1835	MSR_S1_PMON_EVNTSEL0	Package	Uncore SBo 1 Perfmon Event Select for SBo 1 Counter 0
72CH	1836	MSR_S1_PMON_EVNTSEL1	Package	Uncore SBo 1 Perfmon Event Select for SBo 1 Counter 1
72DH	1837	MSR_S1_PMON_EVNTSEL2	Package	Uncore SBo 1 Perfmon Event Select for SBo 1 Counter 2
72EH	1838	MSR_S1_PMON_EVNTSEL3	Package	Uncore SBo 1 Perfmon Event Select for SBo 1 Counter 3
72FH	1839	MSR_S1_PMON_BOX_FILTER	Package	Uncore SBo 1 Perfmon Box-Wide Filter
730H	1840	MSR_S1_PMON_CTR0	Package	Uncore SBo 1 Perfmon Counter 0
731H	1841	MSR_S1_PMON_CTR1	Package	Uncore SBo 1 Perfmon Counter 1
732H	1842	MSR_S1_PMON_CTR2	Package	Uncore SBo 1 Perfmon Counter 2
733H	1843	MSR_S1_PMON_CTR3	Package	Uncore SBo 1 Perfmon Counter 3
734H	1844	MSR_S2_PMON_BOX_CTL	Package	Uncore SBo 2 Perfmon for SBo 2 Box-Wide Control
735H	1845	MSR_S2_PMON_EVNTSEL0	Package	Uncore SBo 2 Perfmon Event Select for SBo 2 Counter 0
736H	1846	MSR_S2_PMON_EVNTSEL1	Package	Uncore SBo 2 Perfmon Event Select for SBo 2 Counter 1
737H	1847	MSR_S2_PMON_EVNTSEL2	Package	Uncore SBo 2 Perfmon Event Select for SBo 2 Counter 2
738H	1848	MSR_S2_PMON_EVNTSEL3	Package	Uncore SBo 2 Perfmon Event Select for SBo 2 Counter 3
739H	1849	MSR_S2_PMON_BOX_FILTER	Package	Uncore SBo 2 Perfmon Box-Wide Filter
73AH	1850	MSR_S2_PMON_CTR0	Package	Uncore SBo 2 Perfmon Counter 0
73BH	1851	MSR_S2_PMON_CTR1	Package	Uncore SBo 2 Perfmon Counter 1
73CH	1852	MSR_S2_PMON_CTR2	Package	Uncore SBo 2 Perfmon Counter 2
73DH	1853	MSR_S2_PMON_CTR3	Package	Uncore SBo 2 Perfmon Counter 3
73EH	1854	MSR_S3_PMON_BOX_CTL	Package	Uncore SBo 3 Perfmon for SBo 3 Box-Wide Control
73FH	1855	MSR_S3_PMON_EVNTSEL0	Package	Uncore SBo 3 Perfmon Event Select for SBo 3 Counter 0
740H	1856	MSR_S3_PMON_EVNTSEL1	Package	Uncore SBo 3 Perfmon Event Select for SBo 3 Counter 1
741H	1857	MSR_S3_PMON_EVNTSEL2	Package	Uncore SBo 3 Perfmon Event Select for SBo 3 Counter 2
742H	1858	MSR_S3_PMON_EVNTSEL3	Package	Uncore SBo 3 Perfmon Event Select for SBo 3 Counter 3
743H	1859	MSR_S3_PMON_BOX_FILTER	Package	Uncore SBo 3 Perfmon Box-Wide Filter
744H	1860	MSR_S3_PMON_CTR0	Package	Uncore SBo 3 Perfmon Counter 0
745H	1861	MSR_S3_PMON_CTR1	Package	Uncore SBo 3 Perfmon Counter 1
746H	1862	MSR_S3_PMON_CTR2	Package	Uncore SBo 3 Perfmon Counter 2
747H	1863	MSR_S3_PMON_CTR3	Package	Uncore SBo 3 Perfmon Counter 3

Table 2-33. Uncore PMU MSRs in Intel® Xeon® Processor E5 v3 Family (Contd.)

	ister Iress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
EOOH	3584	MSR_CO_PMON_BOX_CTL	Package	Uncore C-Box O Perfmon for Box-Wide Control
E01H	3585	MSR_CO_PMON_EVNTSELO	Package	Uncore C-Box 0 Perfmon Event Select for C-Box 0 Counter 0
E02H	3586	MSR_CO_PMON_EVNTSEL1	Package	Uncore C-Box 0 Perfmon Event Select for C-Box 0 Counter 1
E03H	3587	MSR_CO_PMON_EVNTSEL2	Package	Uncore C-Box 0 Perfmon Event Select for C-Box 0 Counter 2
E04H	3588	MSR_CO_PMON_EVNTSEL3	Package	Uncore C-Box 0 Perfmon Event Select for C-Box 0 Counter 3
E05H	3589	MSR_CO_PMON_BOX_FILTERO	Package	Uncore C-Box 0 Perfmon Box Wide Filter 0
E06H	3590	MSR_CO_PMON_BOX_FILTER1	Package	Uncore C-Box 0 Perfmon Box Wide Filter 1
E07H	3591	MSR_CO_PMON_BOX_STATUS	Package	Uncore C-Box 0 Perfmon Box Wide Status
E08H	3592	MSR_CO_PMON_CTRO	Package	Uncore C-Box 0 Perfmon Counter 0
E09H	3593	MSR_CO_PMON_CTR1	Package	Uncore C-Box 0 Perfmon Counter 1
EOAH	3594	MSR_CO_PMON_CTR2	Package	Uncore C-Box O Perfmon Counter 2
EOBH	3595	MSR_CO_PMON_CTR3	Package	Uncore C-Box O Perfmon Counter 3
E10H	3600	MSR_C1_PMON_BOX_CTL	Package	Uncore C-Box 1 Perfmon for Box-Wide Control
E11H	3601	MSR_C1_PMON_EVNTSEL0	Package	Uncore C-Box 1 Perfmon Event Select for C-Box 1 Counter 0
E12H	3602	MSR_C1_PMON_EVNTSEL1	Package	Uncore C-Box 1 Perfmon Event Select for C-Box 1 Counter 1
E13H	3603	MSR_C1_PMON_EVNTSEL2	Package	Uncore C-Box 1 Perfmon Event Select for C-Box 1 Counter 2
E14H	3604	MSR_C1_PMON_EVNTSEL3	Package	Uncore C-Box 1 Perfmon Event Select for C-Box 1 Counter 3
E15H	3605	MSR_C1_PMON_BOX_FILTER0	Package	Uncore C-Box 1 Perfmon Box Wide Filter 0
E16H	3606	MSR_C1_PMON_BOX_FILTER1	Package	Uncore C-Box 1 Perfmon Box Wide Filter1
E17H	3607	MSR_C1_PMON_BOX_STATUS	Package	Uncore C-Box 1 Perfmon Box Wide Status
E18H	3608	MSR_C1_PMON_CTR0	Package	Uncore C-Box 1 Perfmon Counter 0
E19H	3609	MSR_C1_PMON_CTR1	Package	Uncore C-Box 1 Perfmon Counter 1
E1AH	3610	MSR_C1_PMON_CTR2	Package	Uncore C-Box 1 Perfmon Counter 2
E1BH	3611	MSR_C1_PMON_CTR3	Package	Uncore C-Box 1 Perfmon Counter 3
E20H	3616	MSR_C2_PMON_BOX_CTL	Package	Uncore C-Box 2 Perfmon for Box-Wide Control
E21H	3617	MSR_C2_PMON_EVNTSEL0	Package	Uncore C-Box 2 Perfmon Event Select for C-Box 2 Counter 0
E22H	3618	MSR_C2_PMON_EVNTSEL1	Package	Uncore C-Box 2 Perfmon Event Select for C-Box 2 Counter 1
E23H	3619	MSR_C2_PMON_EVNTSEL2	Package	Uncore C-Box 2 Perfmon Event Select for C-Box 2 Counter 2
E24H	3620	MSR_C2_PMON_EVNTSEL3	Package	Uncore C-Box 2 Perfmon Event select for C-Box 2 Counter 3

Table 2-33. Uncore PMU MSRs in Intel® Xeon® Processor E5 v3 Family (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
E25H	3621	MSR_C2_PMON_BOX_FILTER0	Package	Uncore C-Box 2 Perfmon Box Wide Filter 0
E26H	3622	MSR_C2_PMON_BOX_FILTER1	Package	Uncore C-Box 2 Perfmon Box Wide Filter1
E27H	3623	MSR_C2_PMON_BOX_STATUS	Package	Uncore C-Box 2 Perfmon Box Wide Status
E28H	3624	MSR_C2_PMON_CTR0	Package	Uncore C-Box 2 Perfmon Counter 0
E29H	3625	MSR_C2_PMON_CTR1	Package	Uncore C-Box 2 Perfmon Counter 1
E2AH	3626	MSR_C2_PMON_CTR2	Package	Uncore C-Box 2 Perfmon Counter 2
E2BH	3627	MSR_C2_PMON_CTR3	Package	Uncore C-Box 2 Perfmon Counter 3
E30H	3632	MSR_C3_PMON_BOX_CTL	Package	Uncore C-Box 3 Perfmon for Box-Wide Control
E31H	3633	MSR_C3_PMON_EVNTSEL0	Package	Uncore C-Box 3 Perfmon Event Select for C-Box 3 Counter 0
E32H	3634	MSR_C3_PMON_EVNTSEL1	Package	Uncore C-Box 3 Perfmon Event Select for C-Box 3 Counter 1
E33H	3635	MSR_C3_PMON_EVNTSEL2	Package	Uncore C-Box 3 Perfmon Event Select for C-Box 3 Counter 2
E34H	3636	MSR_C3_PMON_EVNTSEL3	Package	Uncore C-Box 3 Perfmon Event Select for C-Box 3 Counter 3
E35H	3637	MSR_C3_PMON_BOX_FILTER0	Package	Uncore C-Box 3 Perfmon Box Wide Filter 0
E36H	3638	MSR_C3_PMON_BOX_FILTER1	Package	Uncore C-Box 3 Perfmon Box Wide Filter1
E37H	3639	MSR_C3_PMON_BOX_STATUS	Package	Uncore C-Box 3 Perfmon Box Wide Status
E38H	3640	MSR_C3_PMON_CTR0	Package	Uncore C-Box 3 Perfmon Counter 0
E39H	3641	MSR_C3_PMON_CTR1	Package	Uncore C-Box 3 Perfmon Counter 1
ЕЗАН	3642	MSR_C3_PMON_CTR2	Package	Uncore C-Box 3 Perfmon Counter 2
E3BH	3643	MSR_C3_PMON_CTR3	Package	Uncore C-Box 3 Perfmon Counter 3
E40H	3648	MSR_C4_PMON_BOX_CTL	Package	Uncore C-Box 4 Perfmon for Box-Wide Control
E41H	3649	MSR_C4_PMON_EVNTSEL0	Package	Uncore C-Box 4 Perfmon Event Select for C-Box 4 Counter 0
E42H	3650	MSR_C4_PMON_EVNTSEL1	Package	Uncore C-Box 4 Perfmon Event Select for C-Box 4 Counter 1
E43H	3651	MSR_C4_PMON_EVNTSEL2	Package	Uncore C-Box 4 Perfmon Event Select for C-Box 4 Counter 2
E44H	3652	MSR_C4_PMON_EVNTSEL3	Package	Uncore C-Box 4 Perfmon Event Select for C-Box 4 Counter 3
E45H	3653	MSR_C4_PMON_BOX_FILTER0	Package	Uncore C-Box 4 Perfmon Box Wide Filter 0
E46H	3654	MSR_C4_PMON_BOX_FILTER1	Package	Uncore C-Box 4 Perfmon Box Wide Filter1
E47H	3655	MSR_C4_PMON_BOX_STATUS	Package	Uncore C-Box 4 Perfmon Box Wide Status
E48H	3656	MSR_C4_PMON_CTR0	Package	Uncore C-Box 4 Perfmon Counter 0
E49H	3657	MSR_C4_PMON_CTR1	Package	Uncore C-Box 4 Perfmon Counter 1
E4AH	3658	MSR_C4_PMON_CTR2	Package	Uncore C-Box 4 Perfmon Counter 2
E4BH	3659	MSR_C4_PMON_CTR3	Package	Uncore C-Box 4 Perfmon Counter 3

Table 2-33. Uncore PMU MSRs in Intel® Xeon® Processor E5 v3 Family (Contd.)

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
E50H	3664	MSR_C5_PMON_BOX_CTL	Package	Uncore C-Box 5 Perfmon for Box-Wide Control
E51H	3665	MSR_C5_PMON_EVNTSEL0	Package	Uncore C-Box 5 Perfmon Event Select for C-Box 5 Counter 0
E52H	3666	MSR_C5_PMON_EVNTSEL1	Package	Uncore C-Box 5 Perfmon Event Select for C-Box 5 Counter 1
E53H	3667	MSR_C5_PMON_EVNTSEL2	Package	Uncore C-Box 5 Perfmon Event Select for C-Box 5 Counter 2
E54H	3668	MSR_C5_PMON_EVNTSEL3	Package	Uncore C-Box 5 Perfmon Event Select for C-Box 5 Counter 3
E55H	3669	MSR_C5_PMON_BOX_FILTER0	Package	Uncore C-Box 5 Perfmon Box Wide Filter 0
E56H	3670	MSR_C5_PMON_BOX_FILTER1	Package	Uncore C-Box 5 Perfmon Box Wide Filter 1
E57H	3671	MSR_C5_PMON_BOX_STATUS	Package	Uncore C-Box 5 Perfmon Box Wide Status
E58H	3672	MSR_C5_PMON_CTR0	Package	Uncore C-Box 5 Perfmon Counter 0
E59H	3673	MSR_C5_PMON_CTR1	Package	Uncore C-Box 5 Perfmon Counter 1
E5AH	3674	MSR_C5_PMON_CTR2	Package	Uncore C-Box 5 Perfmon Counter 2
E5BH	3675	MSR_C5_PMON_CTR3	Package	Uncore C-Box 5 Perfmon Counter 3
E60H	3680	MSR_C6_PMON_BOX_CTL	Package	Uncore C-Box 6 Perfmon for Box-Wide Control
E61H	3681	MSR_C6_PMON_EVNTSEL0	Package	Uncore C-Box 6 Perfmon Event Select for C-Box 6 Counter 0
E62H	3682	MSR_C6_PMON_EVNTSEL1	Package	Uncore C-Box 6 Perfmon Event Select for C-Box 6 Counter 1
E63H	3683	MSR_C6_PMON_EVNTSEL2	Package	Uncore C-Box 6 Perfmon Event Select for C-Box 6 Counter 2
E64H	3684	MSR_C6_PMON_EVNTSEL3	Package	Uncore C-Box 6 Perfmon Event Select for C-Box 6 Counter 3
E65H	3685	MSR_C6_PMON_BOX_FILTER0	Package	Uncore C-Box 6 Perfmon Box Wide Filter 0
E66H	3686	MSR_C6_PMON_BOX_FILTER1	Package	Uncore C-Box 6 Perfmon Box Wide Filter 1
E67H	3687	MSR_C6_PMON_BOX_STATUS	Package	Uncore C-Box 6 Perfmon Box Wide Status
E68H	3688	MSR_C6_PMON_CTR0	Package	Uncore C-Box 6 Perfmon Counter 0
E69H	3689	MSR_C6_PMON_CTR1	Package	Uncore C-Box 6 Perfmon Counter 1
E6AH	3690	MSR_C6_PMON_CTR2	Package	Uncore C-Box 6 Perfmon Counter 2
E6BH	3691	MSR_C6_PMON_CTR3	Package	Uncore C-Box 6 Perfmon Counter 3
E70H	3696	MSR_C7_PMON_BOX_CTL	Package	Uncore C-Box 7 Perfmon for Box-Wide Control
E71H	3697	MSR_C7_PMON_EVNTSEL0	Package	Uncore C-Box 7 Perfmon Event Select for C-Box 7 Counter 0
E72H	3698	MSR_C7_PMON_EVNTSEL1	Package	Uncore C-Box 7 Perfmon Event Select for C-Box 7 Counter 1
E73H	3699	MSR_C7_PMON_EVNTSEL2	Package	Uncore C-Box 7 Perfmon Event Select for C-Box 7 Counter 2
E74H	3700	MSR_C7_PMON_EVNTSEL3	Package	Uncore C-Box 7 Perfmon Event Select for C-Box 7 Counter 3

Table 2-33. Uncore PMU MSRs in Intel® Xeon® Processor E5 v3 Family (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
E75H	3701	MSR_C7_PMON_BOX_FILTER0	Package	Uncore C-Box 7 Perfmon Box Wide Filter 0
E76H	3702	MSR_C7_PMON_BOX_FILTER1	Package	Uncore C-Box 7 Perfmon Box Wide Filter 1
E77H	3703	MSR_C7_PMON_BOX_STATUS	Package	Uncore C-Box 7 Perfmon Box Wide Status
E78H	3704	MSR_C7_PMON_CTR0	Package	Uncore C-Box 7 Perfmon Counter 0
E79H	3705	MSR_C7_PMON_CTR1	Package	Uncore C-Box 7 Perfmon Counter 1
E7AH	3706	MSR_C7_PMON_CTR2	Package	Uncore C-Box 7 Perfmon Counter 2
E7BH	3707	MSR_C7_PMON_CTR3	Package	Uncore C-Box 7 Perfmon Counter 3
E80H	3712	MSR_C8_PMON_BOX_CTL	Package	Uncore C-Box 8 Perfmon Local Box Wide Control
E81H	3713	MSR_C8_PMON_EVNTSEL0	Package	Uncore C-Box 8 Perfmon Event Select for C-Box 8 Counter 0
E82H	3714	MSR_C8_PMON_EVNTSEL1	Package	Uncore C-Box 8 Perfmon Event Select for C-Box 8 Counter 1
E83H	3715	MSR_C8_PMON_EVNTSEL2	Package	Uncore C-Box 8 Perfmon Event Select for C-Box 8 Counter 2
E84H	3716	MSR_C8_PMON_EVNTSEL3	Package	Uncore C-Box 8 Perfmon Event Select for C-Box 8 Counter 3
E85H	3717	MSR_C8_PMON_BOX_FILTER0	Package	Uncore C-Box 8 Perfmon Box Wide Filter 0
E86H	3718	MSR_C8_PMON_BOX_FILTER1	Package	Uncore C-Box 8 Perfmon Box Wide Filter 1
E87H	3719	MSR_C8_PMON_BOX_STATUS	Package	Uncore C-Box 8 Perfmon Box Wide Status
E88H	3720	MSR_C8_PMON_CTR0	Package	Uncore C-Box 8 Perfmon Counter 0
E89H	3721	MSR_C8_PMON_CTR1	Package	Uncore C-Box 8 Perfmon Counter 1
E8AH	3722	MSR_C8_PMON_CTR2	Package	Uncore C-Box 8 Perfmon Counter 2
E8BH	3723	MSR_C8_PMON_CTR3	Package	Uncore C-Box 8 Perfmon Counter 3
E90H	3728	MSR_C9_PMON_BOX_CTL	Package	Uncore C-Box 9 Perfmon Local Box Wide Control
E91H	3729	MSR_C9_PMON_EVNTSEL0	Package	Uncore C-Box 9 Perfmon Event Select for C-Box 9 Counter 0
E92H	3730	MSR_C9_PMON_EVNTSEL1	Package	Uncore C-Box 9 Perfmon Event Select for C-Box 9 Counter 1
E93H	3731	MSR_C9_PMON_EVNTSEL2	Package	Uncore C-Box 9 Perfmon Event Select for C-Box 9 Counter 2
E94H	3732	MSR_C9_PMON_EVNTSEL3	Package	Uncore C-Box 9 Perfmon Event Select for C-Box 9 Counter 3
E95H	3733	MSR_C9_PMON_BOX_FILTER0	Package	Uncore C-Box 9 Perfmon Box Wide Filter 0
E96H	3734	MSR_C9_PMON_BOX_FILTER1	Package	Uncore C-Box 9 Perfmon Box Wide Filter 1
E97H	3735	MSR_C9_PMON_BOX_STATUS	Package	Uncore C-Box 9 Perfmon Box Wide Status
E98H	3736	MSR_C9_PMON_CTR0	Package	Uncore C-Box 9 Perfmon Counter 0
E99H	3737	MSR_C9_PMON_CTR1	Package	Uncore C-Box 9 Perfmon Counter 1
E9AH	3738	MSR_C9_PMON_CTR2	Package	Uncore C-Box 9 Perfmon Counter 2
E9BH	3739	MSR_C9_PMON_CTR3	Package	Uncore C-Box 9 Perfmon Counter 3

Table 2-33. Uncore PMU MSRs in Intel® Xeon® Processor E5 v3 Family (Contd.)

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
EAOH	3744	MSR_C10_PMON_BOX_CTL	Package	Uncore C-Box 10 Perfmon Local Box Wide Control
EA1H	3745	MSR_C10_PMON_EVNTSEL0	Package	Uncore C-Box 10 Perfmon Event Select for C-Box 10 Counter 0
EA2H	3746	MSR_C10_PMON_EVNTSEL1	Package	Uncore C-Box 10 Perfmon Event Select for C-Box 10 Counter 1
ЕАЗН	3747	MSR_C10_PMON_EVNTSEL2	Package	Uncore C-Box 10 Perfmon Event Select for C-Box 10 Counter 2
EA4H	3748	MSR_C10_PMON_EVNTSEL3	Package	Uncore C-Box 10 Perfmon Event Select for C-Box 10 Counter 3
EA5H	3749	MSR_C10_PMON_BOX_FILTER0	Package	Uncore C-Box 10 Perfmon Box Wide Filter 0
EA6H	3750	MSR_C10_PMON_BOX_FILTER1	Package	Uncore C-Box 10 Perfmon Box Wide Filter 1
EA7H	3751	MSR_C10_PMON_BOX_STATUS	Package	Uncore C-Box 10 Perfmon Box Wide Status
EA8H	3752	MSR_C10_PMON_CTR0	Package	Uncore C-Box 10 Perfmon Counter 0
EA9H	3753	MSR_C10_PMON_CTR1	Package	Uncore C-Box 10 perfmon Counter 1
EAAH	3754	MSR_C10_PMON_CTR2	Package	Uncore C-Box 10 Perfmon Counter 2
EABH	3755	MSR_C10_PMON_CTR3	Package	Uncore C-Box 10 Perfmon Counter 3
EBOH	3760	MSR_C11_PMON_BOX_CTL	Package	Uncore C-Box 11 Perfmon Local Box Wide Control
EB1H	3761	MSR_C11_PMON_EVNTSEL0	Package	Uncore C-Box 11 Perfmon Event Select for C-Box 11 Counter 0
EB2H	3762	MSR_C11_PMON_EVNTSEL1	Package	Uncore C-Box 11 Perfmon Event Select for C-Box 11 Counter 1
EB3H	3763	MSR_C11_PMON_EVNTSEL2	Package	Uncore C-Box 11 Perfmon Event Select for C-Box 11 Counter 2
EB4H	3764	MSR_C11_PMON_EVNTSEL3	Package	Uncore C-box 11 Perfmon Event Select for C-Box 11 Counter 3
EB5H	3765	MSR_C11_PMON_BOX_FILTER0	Package	Uncore C-Box 11 Perfmon Box Wide Filter 0
EB6H	3766	MSR_C11_PMON_BOX_FILTER1	Package	Uncore C-Box 11 Perfmon Box Wide Filter 1
EB7H	3767	MSR_C11_PMON_BOX_STATUS	Package	Uncore C-Box 11 Perfmon Box Wide Status
EB8H	3768	MSR_C11_PMON_CTR0	Package	Uncore C-Box 11 Perfmon Counter 0
EB9H	3769	MSR_C11_PMON_CTR1	Package	Uncore C-Box 11 Perfmon Counter 1
EBAH	3770	MSR_C11_PMON_CTR2	Package	Uncore C-Box 11 Perfmon Counter 2
EBBH	3771	MSR_C11_PMON_CTR3	Package	Uncore C-Box 11 Perfmon Counter 3
ECOH	3776	MSR_C12_PMON_BOX_CTL	Package	Uncore C-Box 12 Perfmon Local Box Wide Control
EC1H	3777	MSR_C12_PMON_EVNTSEL0	Package	Uncore C-Box 12 Perfmon Event Select for C-Box 12 Counter 0
EC2H	3778	MSR_C12_PMON_EVNTSEL1	Package	Uncore C-Box 12 Perfmon Event Select for C-Box 12 Counter 1
EC3H	3779	MSR_C12_PMON_EVNTSEL2	Package	Uncore C-Box 12 Perfmon Event Select for C-Box 12 Counter 2
EC4H	3780	MSR_C12_PMON_EVNTSEL3	Package	Uncore C-Box 12 Perfmon Event Select for C-Box 12 Counter 3

Table 2-33. Uncore PMU MSRs in Intel® Xeon® Processor E5 v3 Family (Contd.)

	ister Iress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
EC5H	3781	MSR_C12_PMON_BOX_FILTER0	Package	Uncore C-Box 12 Perfmon Box Wide Filter 0
EC6H	3782	MSR_C12_PMON_BOX_FILTER1	Package	Uncore C-Box 12 Perfmon Box Wide Filter 1
EC7H	3783	MSR_C12_PMON_BOX_STATUS	Package	Uncore C-Box 12 Perfmon Box Wide Status
EC8H	3784	MSR_C12_PMON_CTR0	Package	Uncore C-Box 12 Perfmon Counter 0
EC9H	3785	MSR_C12_PMON_CTR1	Package	Uncore C-Box 12 Perfmon Counter 1
ECAH	3786	MSR_C12_PMON_CTR2	Package	Uncore C-Box 12 Perfmon Counter 2
ECBH	3787	MSR_C12_PMON_CTR3	Package	Uncore C-Box 12 Perfmon Counter 3
EDOH	3792	MSR_C13_PMON_BOX_CTL	Package	Uncore C-Box 13 Perfmon local box wide control.
ED1H	3793	MSR_C13_PMON_EVNTSEL0	Package	Uncore C-Box 13 Perfmon Event Select for C-Box 13 Counter 0
ED2H	3794	MSR_C13_PMON_EVNTSEL1	Package	Uncore C-Box 13 Perfmon Event Select for C-Box 13 Counter 1
ED3H	3795	MSR_C13_PMON_EVNTSEL2	Package	Uncore C-Box 13 Perfmon Event Select for C-Box 13 Counter 2
ED4H	3796	MSR_C13_PMON_EVNTSEL3	Package	Uncore C-Box 13 Perfmon Event Select for C-Box 13 Counter 3
ED5H	3797	MSR_C13_PMON_BOX_FILTER0	Package	Uncore C-Box 13 Perfmon Box Wide Filter 0
ED6H	3798	MSR_C13_PMON_BOX_FILTER1	Package	Uncore C-Box 13 Perfmon Box Wide Filter 1
ED7H	3799	MSR_C13_PMON_BOX_STATUS	Package	Uncore C-Box 13 Perfmon Box Wide Status
ED8H	3800	MSR_C13_PMON_CTR0	Package	Uncore C-Box 13 Perfmon Counter 0
ED9H	3801	MSR_C13_PMON_CTR1	Package	Uncore C-Box 13 Perfmon Counter 1
EDAH	3802	MSR_C13_PMON_CTR2	Package	Uncore C-Box 13 Perfmon Counter 2
EDBH	3803	MSR_C13_PMON_CTR3	Package	Uncore C-Box 13 Perfmon Counter 3
EEOH	3808	MSR_C14_PMON_BOX_CTL	Package	Uncore C-Box 14 Perfmon Local Box Wide Control
EE1H	3809	MSR_C14_PMON_EVNTSEL0	Package	Uncore C-Box 14 Perfmon Event Select for C-Box 14 Counter 0
EE2H	3810	MSR_C14_PMON_EVNTSEL1	Package	Uncore C-Box 14 Perfmon Event Select for C-Box 14 Counter 1
EE3H	3811	MSR_C14_PMON_EVNTSEL2	Package	Uncore C-Box 14 Perfmon Event Select for C-Box 14 Counter 2
EE4H	3812	MSR_C14_PMON_EVNTSEL3	Package	Uncore C-Box 14 Perfmon Event Select for C-Box 14 Counter 3
EE5H	3813	MSR_C14_PMON_BOX_FILTER	Package	Uncore C-Box 14 Perfmon Box Wide Filter 0
EE6H	3814	MSR_C14_PMON_BOX_FILTER1	Package	Uncore C-Box 14 Perfmon Box Wide Filter 1
EE7H	3815	MSR_C14_PMON_BOX_STATUS	Package	Uncore C-Box 14 Perfmon Box Wide Status
EE8H	3816	MSR_C14_PMON_CTR0	Package	Uncore C-Box 14 Perfmon Counter 0
EE9H	3817	MSR_C14_PMON_CTR1	Package	Uncore C-Box 14 Perfmon Counter 1
EEAH	3818	MSR_C14_PMON_CTR2	Package	Uncore C-Box 14 Perfmon Counter 2
EEBH	3819	MSR_C14_PMON_CTR3	Package	Uncore C-Box 14 Perfmon Counter 3

Table 2-33. Uncore PMU MSRs in Intel® Xeon® Processor E5 v3 Family (Contd.)

	ister Iress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
EFOH	3824	MSR_C15_PMON_BOX_CTL	Package	Uncore C-Box 15 Perfmon Local Box Wide Control
EF1H	3825	MSR_C15_PMON_EVNTSEL0	Package	Uncore C-Box 15 Perfmon Event Select for C-Box 15 Counter 0
EF2H	3826	MSR_C15_PMON_EVNTSEL1	Package	Uncore C-Box 15 Perfmon Event Select for C-Box 15 Counter 1
EF3H	3827	MSR_C15_PMON_EVNTSEL2	Package	Uncore C-Box 15 Perfmon Event Select for C-Box 15 Counter 2
EF4H	3828	MSR_C15_PMON_EVNTSEL3	Package	Uncore C-Box 15 Perfmon Event Select for C-Box 15 Counter 3
EF5H	3829	MSR_C15_PMON_BOX_FILTER0	Package	Uncore C-Box 15 Perfmon Box Wide Filter 0
EF6H	3830	MSR_C15_PMON_BOX_FILTER1	Package	Uncore C-Box 15 Perfmon Box Wide Filter 1
EF7H	3831	MSR_C15_PMON_BOX_STATUS	Package	Uncore C-Box 15 Perfmon Box Wide Status
EF8H	3832	MSR_C15_PMON_CTR0	Package	Uncore C-Box 15 Perfmon Counter 0
EF9H	3833	MSR_C15_PMON_CTR1	Package	Uncore C-Box 15 Perfmon Counter 1
EFAH	3834	MSR_C15_PMON_CTR2	Package	Uncore C-Box 15 Perfmon Counter 2
EFBH	3835	MSR_C15_PMON_CTR3	Package	Uncore C-Box 15 Perfmon Counter 3
F00H	3840	MSR_C16_PMON_BOX_CTL	Package	Uncore C-Box 16 Perfmon for Box-Wide Control
F01H	3841	MSR_C16_PMON_EVNTSEL0	Package	Uncore C-Box 16 Perfmon Event Select for C-Box 16 Counter 0
F02H	3842	MSR_C16_PMON_EVNTSEL1	Package	Uncore C-Box 16 Perfmon Event Select for C-Box 16 Counter 1
F03H	3843	MSR_C16_PMON_EVNTSEL2	Package	Uncore C-Box 16 Perfmon Event Select for C-Box 16 Counter 2
F04H	3844	MSR_C16_PMON_EVNTSEL3	Package	Uncore C-Box 16 Perfmon Event Select for C-Box 16 Counter 3
F05H	3845	MSR_C16_PMON_BOX_FILTER0	Package	Uncore C-Box 16 Perfmon Box Wide Filter 0
F06H	3846	MSR_C16_PMON_BOX_FILTER1	Package	Uncore C-Box 16 Perfmon Box Wide Filter 1
F07H	3847	MSR_C16_PMON_BOX_STATUS	Package	Uncore C-Box 16 Perfmon Box Wide Status
F08H	3848	MSR_C16_PMON_CTR0	Package	Uncore C-Box 16 Perfmon Counter 0
F09H	3849	MSR_C16_PMON_CTR1	Package	Uncore C-Box 16 Perfmon Counter 1
FOAH	3850	MSR_C16_PMON_CTR2	Package	Uncore C-Box 16 Perfmon Counter 2
FOBH	3851	MSR_C16_PMON_CTR3	Package	Uncore C-Box 16 Perfmon Counter 3
F10H	3856	MSR_C17_PMON_BOX_CTL	Package	Uncore C-Box 17 Perfmon for Box-Wide Control
F11H	3857	MSR_C17_PMON_EVNTSEL0	Package	Uncore C-Box 17 Perfmon Event Select for C-Box 17 Counter 0
F12H	3858	MSR_C17_PMON_EVNTSEL1	Package	Uncore C-Box 17 Perfmon Event Select for C-Box 17 Counter 1
F13H	3859	MSR_C17_PMON_EVNTSEL2	Package	Uncore C-Box 17 Perfmon Event Select for C-Box 17 Counter 2
F14H	3860	MSR_C17_PMON_EVNTSEL3	Package	Uncore C-Box 17 Perfmon Event Select for C-Box 17 Counter 3

_	ister Iress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
F15H	3861	MSR_C17_PMON_BOX_FILTER0	Package	Uncore C-Box 17 Perfmon Box Wide Filter 0
F16H	3862	MSR_C17_PMON_BOX_FILTER1	Package	Uncore C-Box 17 Perfmon Box Wide Filter1
F17H	3863	MSR_C17_PMON_BOX_STATUS	Package	Uncore C-Box 17 Perfmon Box Wide Status
F18H	3864	MSR_C17_PMON_CTR0	Package	Uncore C-Box 17 Perfmon Counter 0
F19H	3865	MSR_C17_PMON_CTR1	Package	Uncore C-Box 17 Perfmon Counter 1
F1AH	3866	MSR_C17_PMON_CTR2	Package	Uncore C-Box 17 Perfmon Counter 2
F1BH	3867	MSR C17 PMON CTR3	Package	Uncore C-Box 17 Perfmon Counter 3

Table 2-33. Uncore PMU MSRs in Intel® Xeon® Processor E5 v3 Family (Contd.)

2.15 MSRS IN INTEL® CORE™ M PROCESSORS AND 5TH GENERATION INTEL CORE PROCESSORS

The Intel[®] Core[™] M-5xxx processors and 5th generation Intel[®] Core[™] Processors, and Intel[®] Xeon[®] Processor E3-1200 v4 family are based on the Broadwell microarchitecture. The Intel[®] Core[™] M-5xxx processors and 5th generation Intel[®] Core[™] Processors have CPUID DisplayFamily_DisplayModel signature 06_3DH. Intel[®] Xeon[®] Processor E3-1200 v4 family and the 5th generation Intel[®] Core[™] Processors have CPUID DisplayFamily_DisplayModel signature 06_47H. Processors with signatures 06_3DH and 06_47H support the MSR interfaces listed in Table 2-20, Table 2-21, Table 2-22, Table 2-25, Table 2-29, Table 2-30, Table 2-34, and Table 2-35. For an MSR listed in Table 2-35 that also appears in the model-specific tables of prior generations, Table 2-35 supersede prior generation tables.

Table 2-34 lists MSRs that are common to processors based on the Broadwell microarchitectures (including CPUID signatures 06_3DH, 06_47H, 06_4FH, and 06_56H).

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
38EH	910	IA32_PERF_GLOBAL_STATUS	Thread	See Table 2-2. See Section 20.6.2.2, "Global Counter Control Facilities."
		0		Ovf_PMC0
		1		Ovf_PMC1
		2		Ovf_PMC2
		3		Ovf_PMC3
		31:4		Reserved
		32		Ovf_FixedCtr0
		33		Ovf_FixedCtr1
		34		Ovf_FixedCtr2
		54:35		Reserved
		55		Trace_ToPA_PMI
				See Section 33.2.7.2, "Table of Physical Addresses (ToPA)."
		60:56		Reserved

Table 2-34 Additional MSRs Common to Processors Based the Broadwell Microarchitectures

Table 2-34. Additional MSRs Common to Processors Based the Broadwell Microarchitectures

	ister Iress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	7		
		61		Ovf_Uncore
		62		Ovf_BufDSSAVE
		63		CondChgd
390H	912	IA32_PERF_GLOBAL_OVF_CTRL	Thread	See Table 2-2. See Section 20.6.2.2, "Global Counter Control Facilities."
		0		Set 1 to clear Ovf_PMCO
		1		Set 1 to clear Ovf_PMC1
		2		Set 1 to clear Ovf_PMC2
		3		Set 1 to clear Ovf_PMC3
		31:4		Reserved
		32		Set 1 to clear Ovf_FixedCtr0
		33		Set 1 to clear Ovf_FixedCtr1
		34		Set 1 to clear Ovf_FixedCtr2
		54:35		Reserved.
		55		Set 1 to clear Trace_ToPA_PMI.
				See Section 33.2.7.2, "Table of Physical Addresses (ToPA)."
		60:56		Reserved
		61		Set 1 to clear Ovf_Uncore
		62		Set 1 to clear Ovf_BufDSSAVE
		63		Set 1 to clear CondChgd
560H	1376	IA32_RTIT_OUTPUT_BASE	THREAD	Trace Output Base Register (R/W)
		6:0		Reserved
		MAXPHYADDR ¹ -1:7		Base physical address.
		63:MAXPHYADDR		Reserved
561H	1377	IA32_RTIT_OUTPUT_MASK_PTRS	THREAD	Trace Output Mask Pointers Register (R/W)
		6:0		Reserved
		31:7		MaskOrTableOffset
		63:32		Output Offset.
570H	1392	IA32_RTIT_CTL	Thread	Trace Control Register (R/W)
		0		TraceEn
		1		Reserved, must be zero.
		2		OS
		3		User
		6:4		Reserved, must be zero.
		7		CR3Filter
		8		ToPA
				Writing 0 will #GP if also setting TraceEn.

Table 2-34. Additional MSRs Common to Processors Based the Broadwell Microarchitectures

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		9		Reserved, must be zero.
		10		TSCEn
		11		DisRETC
		12		Reserved, must be zero.
		13		Reserved; writing 0 will #GP if also setting TraceEn.
		63:14		Reserved, must be zero.
571H	1393	IA32_RTIT_STATUS	Thread	Tracing Status Register (R/W)
		0		Reserved, writes ignored.
		1		ContexEn, writes ignored.
		2		TriggerEn, writes ignored.
		3		Reserved
		4		Error (R/W)
		5		Stopped
		63:6		Reserved, must be zero.
572H	1394	IA32_RTIT_CR3_MATCH	THREAD	Trace Filter CR3 Match Register (R/W)
		4:0		Reserved
		63:5		CR3[63:5] value to match.
620H	1568	MSR UNCORE_RATIO_LIMIT	Package	Uncore Ratio Limit (R/W)
				Out of reset, the min_ratio and max_ratio fields represent the widest possible range of uncore frequencies. Writing to these fields allows software to control the minimum and the maximum frequency that hardware will select.
		63:15		Reserved
		14:8		MIN_RATIO
				Writing to this field controls the minimum possible ratio of the LLC/Ring.
		7		Reserved
		6:0		MAX_RATIO
				This field is used to limit the max ratio of the LLC/Ring.

NOTES:

1. MAXPHYADDR is reported by CPUID.80000008H:EAX[7:0].

Table 2-35 lists MSRs that are specific to Intel Core M processors and 5th Generation Intel Core Processors.

Table 2-35. Additional MSRs Supported by Intel® Core™ M Processors and 5th Generation Intel® Core™ Processors

Regi Add		Register Name	Scope	Bit Description
Hex	Dec			
E2H	226	MSR_PKG_CST_CONFIG_CONTROL	Соге	C-State Configuration Control (R/W)
				Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-states.
				See http://biosbits.org.
		3:0		Package C-State Limit (R/W)
				Specifies the lowest processor-specific C-state code name (consuming the least power) for the package. The default is set as factory-configured package C-state limit.
				The following C-state code name encodings are supported:
				0000b: CO/C1 (no package C-state support)
				0001b: C2
				0010b: C3
				0011b: C6
				0100b: C7 0101b: C7s
				0110b: C8
				0111b: C9
				1000b: C10
		9:4		Reserved
		10		I/O MWAIT Redirection Enable (R/W)
		14:11		Reserved
		15		CFG Lock (R/WO)
		24:16		Reserved
		25		C3 State Auto Demotion Enable (R/W)
		26		C1 State Auto Demotion Enable (R/W)
		27		Enable C3 Undemotion (R/W)
		28		Enable C1 Undemotion (R/W)
		29		Enable Package C-State Auto-Demotion (R/W)
		30		Enable Package C-State Undemotion (R/W)
		63:31		Reserved
1ADH	429	MSR_TURBO_RATIO_LIMIT	Package	Maximum Ratio Limit of Turbo Mode
				R/O if MSR_PLATFORM_INFO.[28] = 0.
				R/W if MSR_PLATFORM_INFO.[28] = 1.
		7:0	Package	Maximum Ratio Limit for 1C
				Maximum turbo ratio limit of 1 core active.
		15:8	Package	Maximum Ratio Limit for 2C
				Maximum turbo ratio limit of 2 core active.

Table 2-35. Additional MSRs Supported by Intel® Core™ M Processors and 5th Generation Intel® Core™ Processors

Register Address		Register Name	Scope	Bit Description
Hex	Dec			
		23:16	Package	Maximum Ratio Limit for 3C Maximum turbo ratio limit of 3 core active.
		31:24	Package	Maximum Ratio Limit for 4C Maximum turbo ratio limit of 4 core active.
		39:32	Package	Maximum Ratio Limit for 5C Maximum turbo ratio limit of 5core active.
		47:40	Package	Maximum Ratio Limit for 6C Maximum turbo ratio limit of 6core active.
		63:48		Reserved
639H	1593	MSR_PPO_ENERGY_STATUS	Package	PPO Energy Status (R/O) See Section 15.10.4, "PPO/PP1 RAPL Domains."

See Table 2-20, Table 2-21, Table 2-22, Table 2-25, Table 2-29, Table 2-30, Table 2-34 for other MSR definitions applicable to processors with CPUID signature 06_3DH.

2.16 MSRS IN INTEL® XEON® PROCESSORS E5 V4 FAMILY

The MSRs listed in Table 2-36 are available and common to Intel[®] Xeon[®] Processor D product Family (CPUID DisplayFamily_DisplayModel = 06_56H) and to Intel Xeon processors E5 v4, E7 v4 families (CPUID DisplayFamily_DisplayModel = 06_4FH). They are based on the Broadwell microarchitecture.

See Section 2.16.1 for lists of tables of MSRs that are supported by Intel[®] Xeon[®] Processor D Family.

Table 2-36. Additional MSRs Common to Intel® Xeon® Processor D and Intel Xeon Processors E5 v4 Family Based on the Broadwell Microarchitecture

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
4EH	78	IA32_PPIN_CTL (MSR_PPIN_CTL)	Package	Protected Processor Inventory Number Enable Control (R/W)
		0		LockOut (R/W0)
				See Table 2-2.
		1		Enable_PPIN (R/W)
				See Table 2-2.
		63:2		Reserved
4FH	79	IA32_PPIN (MSR_PPIN)	Package	Protected Processor Inventory Number (R/0)
		63:0		Protected Processor Inventory Number (R/O)
				See Table 2-2.
CEH	206	MSR_PLATFORM_INFO	Package	Platform Information
				Contains power management and other model specific features enumeration. See http://biosbits.org.

Table 2-36. Additional MSRs Common to Intel® Xeon® Processor D and Intel Xeon Processors E5 v4 Family Based on the Broadwell Microarchitecture

Regi Add	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	7		
		7:0		Reserved
		15:8	Package	Maximum Non-Turbo Ratio (R/O)
				See Table 2-26.
		22:16		Reserved.
		23	Package	PPIN_CAP (R/O)
				See Table 2-26.
		27:24		Reserved
		28	Package	Programmable Ratio Limit for Turbo Mode (R/O) See Table 2-26.
		29	Package	Programmable TDP Limit for Turbo Mode (R/O) See Table 2-26.
		30	Package	Programmable TJ OFFSET (R/O)
				See Table 2-26.
		39:31		Reserved
		47:40	Package	Maximum Efficiency Ratio (R/O)
				See Table 2-26.
		63:48		Reserved
E2H	226	MSR_PKG_CST_CONFIG_CONTROL	Core	C-State Configuration Control (R/W)
				Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-states.
				See http://biosbits.org.
		2:0		Package C-State Limit (R/W)
				Specifies the lowest processor-specific C-state code name (consuming the least power) for the package. The default is set as factory-configured package C-state limit.
				The following C-state code name encodings are supported:
				000b: CO/C1 (no package C-state support)
				001b: C2
				010b: C6 (non-retention)
				011b: C6 (retention)
				111b: No Package C state limits. All C states supported by the processor are available.
		9:3		Reserved
		10		I/O MWAIT Redirection Enable (R/W)
		14:11		Reserved
		15		CFG Lock (R/WO)

Table 2-36. Additional MSRs Common to Intel® Xeon® Processor D and Intel Xeon Processors E5 v4 Family Based on the Broadwell Microarchitecture

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	1		
		16		Automatic C-State Conversion Enable (R/W) If 1, the processor will convert HALT or MWAT(C1) to MWAIT(C6).
		24:17		Reserved
		25		C3 State Auto Demotion Enable (R/W)
		26		C1 State Auto Demotion Enable (R/W)
		27		Enable C3 Undemotion (R/W)
		28		Enable C1 Undemotion (R/W)
		29		Package C State Demotion Enable (R/W)
		30		Package C State UnDemotion Enable (R/W)
		63:31		Reserved
179H	377	IA32_MCG_CAP	Thread	Global Machine Check Capability (R/O)
		7:0		Count
		8		MCG_CTL_P
		9		MCG_EXT_P
		10		MCP_CMCI_P
		11		MCG_TES_P
		15:12		Reserved
		23:16		MCG_EXT_CNT
		24		MCG_SER_P
		25		MCG_EM_P
		26		MCG_ELOG_P
		63:27	Thread	Reserved
17DH	381	MSR_SMM_MCA_CAP	Thread	Enhanced SMM Capabilities (SMM-RO) Reports SMM capability Enhancement. Accessible only while in SMM.
		57:0		Reserved
		58		SMM_Code_Access_Chk (SMM-RO)
				If set to 1, indicates that the SMM code access restriction is supported and a host-space interface available to SMM handler.
		59		Long_Flow_Indication (SMM-RO)
				If set to 1, indicates that the SMM long flow indicator is supported and a host-space interface available to SMM handler.
		63:60		Reserved
19CH	412	IA32_THERM_STATUS	Core	Thermal Monitor Status (R/W) See Table 2-2.

Table 2-36. Additional MSRs Common to Intel® Xeon® Processor D and Intel Xeon Processors E5 v4 Family Based on the Broadwell Microarchitecture

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		0		Thermal Status (R/O) See Table 2-2.
		1		Thermal Status Log (R/WC0) See Table 2-2.
		2		PROTCHOT # or FORCEPR# Status (R/O) See Table 2-2.
		3		PROTCHOT # or FORCEPR# Log (R/WCO) See Table 2-2.
		4		Critical Temperature Status (R/0) See Table 2-2.
		5		Critical Temperature Status Log (R/WCO) See Table 2-2.
		6		Thermal Threshold #1 Status (R/O) See Table 2-2.
		7		Thermal Threshold #1 Log (R/WC0) See Table 2-2.
		8		Thermal Threshold #2 Status (R/O) See Table 2-2.
		9		Thermal Threshold #2 Log (R/WC0) See Table 2-2.
		10		Power Limitation Status (R/O) See Table 2-2.
		11		Power Limitation Log (R/WC0) See Table 2-2.
		12		Current Limit Status (R/O) See Table 2-2.
		13		Current Limit Log (R/WC0) See Table 2-2.
		14		Cross Domain Limit Status (R/O) See Table 2-2.
		15		Cross Domain Limit Log (R/WCO) See Table 2-2.
		22:16		Digital Readout (R/O) See Table 2-2.
		26:23		Reserved
		30:27		Resolution in Degrees Celsius (R/O) See Table 2-2.

Table 2-36. Additional MSRs Common to Intel® Xeon® Processor D and Intel Xeon Processors E5 v4 Family Based on the Broadwell Microarchitecture

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		31		Reading Valid (R/O) See Table 2-2.
		63:32		Reserved
1A2H	418	MSR_TEMPERATURE_TARGET	Package	Temperature Target
		15:0		Reserved
		23:16		Temperature Target (R/O) See Table 2-26.
		27:24		TCC Activation Offset (R/W) See Table 2-26.
		63:28		Reserved.
1ADH	429	MSR_TURBO_RATIO_LIMIT	Package	Maximum Ratio Limit of Turbo Mode R/O if MSR_PLATFORM_INFO.[28] = 0. R/W if MSR_PLATFORM_INFO.[28] = 1.
		7:0	Package	Maximum Ratio Limit for 1C
		15:8	Package	Maximum Ratio Limit for 2C
		23:16	Package	Maximum Ratio Limit for 3C
		31:24	Package	Maximum Ratio Limit for 4C
		39:32	Package	Maximum Ratio Limit for 5C
		47:40	Package	Maximum Ratio Limit for 6C
		55:48	Package	Maximum Ratio Limit for 7C
		63:56	Package	Maximum Ratio Limit for 8C
1AEH	430	MSR_TURBO_RATIO_LIMIT1	Package	Maximum Ratio Limit of Turbo Mode
				R/O if MSR_PLATFORM_INFO.[28] = 0.
				R/W if MSR_PLATFORM_INFO.[28] = 1.
		7:0	Package	Maximum Ratio Limit for 9C
		15:8	Package	Maximum Ratio Limit for 10C
		23:16	Package	Maximum Ratio Limit for 11C
		31:24	Package	Maximum Ratio Limit for 12C
		39:32	Package	Maximum Ratio Limit for 13C
		47:40	Package	Maximum Ratio Limit for 14C
		55:48	Package	Maximum Ratio Limit for 15C
		63:56	Package	Maximum Ratio Limit for 16C
606H	1542	MSR_RAPL_POWER_UNIT	Package	Unit Multipliers Used in RAPL Interfaces (R/O)
		3:0	Package	Power Units See Section 15.10.1, "RAPL Interfaces."
		7:4	Package	Reserved

Table 2-36. Additional MSRs Common to Intel® Xeon® Processor D and Intel Xeon Processors E5 v4 Family Based on the Broadwell Microarchitecture

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		12:8	Package	Energy Status Units Energy related information (in Joules) is based on the multiplier, 1/2^ESU; where ESU is an unsigned integer represented by bits 12:8. Default value is 0EH (or 61 micro-joules).
		15:13	Package	Reserved
		19:16	Package	Time Units See Section 15.10.1, "RAPL Interfaces."
		63:20		Reserved
618H	1560	MSR_DRAM_POWER_LIMIT	Package	DRAM RAPL Power Limit Control (R/W) See Section 15.10.5, "DRAM RAPL Domain."
619H	1561	MSR_DRAM_ENERGY_STATUS	Package	DRAM Energy Status (R/O) Energy consumed by DRAM devices.
		31:0		Energy in 15.3 micro-joules. Requires BIOS configuration to enable DRAM RAPL mode 0 (Direct VR).
		63:32		Reserved
61BH	1563	MSR_DRAM_PERF_STATUS	Package	DRAM Performance Throttling Status (R/O) See Section 15.10.5, "DRAM RAPL Domain."
61CH	1564	MSR_DRAM_POWER_INFO	Package	DRAM RAPL Parameters (R/W)
				See Section 15.10.5, "DRAM RAPL Domain."
620H	1568	MSR UNCORE_RATIO_LIMIT	Package	Uncore Ratio Limit (R/W) Out of reset, the min_ratio and max_ratio fields represent the widest possible range of uncore frequencies. Writing to these fields allows software to control the minimum and the maximum frequency that hardware will select.
		63:15		Reserved
		14:8		MIN_RATIO Writing to this field controls the minimum possible ratio of the LLC/Ring.
		7		Reserved
		6:0		MAX_RATIO This field is used to limit the max ratio of the LLC/Ring.
639H	1593	MSR_PPO_ENERGY_STATUS	Package	Reserved (R/O) Reads return 0.
690H	1680	MSR_CORE_PERF_LIMIT_REASONS	Package	Indicator of Frequency Clipping in Processor Cores (R/W)
				(Frequency refers to processor core frequency.)

Table 2-36. Additional MSRs Common to Intel® Xeon® Processor D and Intel Xeon Processors E5 v4 Family Based on the Broadwell Microarchitecture

Regi Addı		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		0		PROCHOT Status (R0)
				When set, processor core frequency is reduced below the operating system request due to assertion of external PROCHOT.
		1		Thermal Status (R0)
				When set, frequency is reduced below the operating system request due to a thermal event.
		2		Power Budget Management Status (R0)
				When set, frequency is reduced below the operating system request due to PBM limit.
		3		Platform Configuration Services Status (R0)
				When set, frequency is reduced below the operating system request due to PCS limit.
		4		Reserved
		5		Autonomous Utilization-Based Frequency Control Status (RO)
				When set, frequency is reduced below the operating system request because the processor has detected that utilization is low.
		6		VR Therm Alert Status (R0)
				When set, frequency is reduced below the operating system request due to a thermal alert from the Voltage Regulator.
		7		Reserved
		8		Electrical Design Point Status (R0)
				When set, frequency is reduced below the operating system request due to electrical design point constraints (e.g., maximum electrical current consumption).
		9		Reserved
		10		Multi-Core Turbo Status (RO)
				When set, frequency is reduced below the operating system request due to Multi-Core Turbo limits.
		12:11		Reserved
		13		Core Frequency P1 Status (R0)
				When set, frequency is reduced below max non-turbo P1.
		14		Core Max N-Core Turbo Frequency Limiting Status (R0)
				When set, frequency is reduced below max n-core turbo frequency.
		15		Core Frequency Limiting Status (R0)
				When set, frequency is reduced below the operating system request.

Table 2-36. Additional MSRs Common to Intel® Xeon® Processor D and Intel Xeon Processors E5 v4 Family Based on the Broadwell Microarchitecture

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		16		PROCHOT Log When set, indicates that the PROCHOT Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.
		17		Thermal Log When set, indicates that the Thermal Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.
		18		Power Budget Management Log When set, indicates that the PBM Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.
		19		Platform Configuration Services Log When set, indicates that the PCS Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.
		20		Reserved
		21		Autonomous Utilization-Based Frequency Control Log When set, indicates that the AUBFC Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.
		22		VR Therm Alert Log When set, indicates that the VR Therm Alert Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.
		23		Reserved
		24		Electrical Design Point Log When set, indicates that the EDP Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.
		25		Reserved
		26		Multi-Core Turbo Log When set, indicates that the Multi-Core Turbo Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.

Table 2-36. Additional MSRs Common to Intel® Xeon® Processor D and Intel Xeon Processors E5 v4 Family Based on the Broadwell Microarchitecture

ister ress	Register Name / Bit Fields	Scope	Bit Description
Dec			
	28:27		Reserved
	29		Core Frequency P1 Log
			When set, indicates that the Core Frequency P1 Status bit has asserted since the log bit was last cleared.
			This log bit will remain set until cleared by software writing 0.
	30		Core Max N-Core Turbo Frequency Limiting Log
			When set, indicates that the Core Max n-core Turbo Frequency Limiting Status bit has asserted since the log bit was last cleared.
			This log bit will remain set until cleared by software writing 0.
	31		Core Frequency Limiting Log
			When set, indicates that the Core Frequency Limiting Status bit has asserted since the log bit was last cleared.
			This log bit will remain set until cleared by software writing 0.
	63:32		Reserved
1904	IA32_PM_ENABLE	Package	See Section 15.4.2, "Enabling HWP".
1905	IA32_HWP_CAPABILITIES	Thread	See Section 15.4.3, "HWP Performance Range and Dynamic Capabilities".
1908	IA32_HWP_REQUEST	Thread	See Section 15.4.4, "Managing HWP".
	7:0		Minimum Performance (R/W)
	15:8		Maximum Performance (R/W)
	23:16		Desired Performance (R/W)
	63:24		Reserved
1911	IA32_HWP_STATUS	Thread	See Section 15.4.5, "HWP Feedback".
	1:0		Reserved
	2		Excursion to Minimum (R/0)
	63:3		Reserved
3213	IA32_QM_EVTSEL	Thread	Monitoring Event Select Register (R/W) If CPUID.(EAX=07H, ECX=0):EBX.RDT-M[bit 12] = 1.
	7:0		EventID (R/W) Event encoding: 0x00: No monitoring. 0x01: L3 occupancy monitoring. 0x02: Total memory bandwidth monitoring. 0x03: Local memory bandwidth monitoring.
	1904 1905 1908	Dec 28:27 29 30 30 31 31 31 31 31 31	Dec 28:27

Table 2-36. Additional MSRs Common to Intel® Xeon® Processor D and Intel Xeon Processors E5 v4 Family Based on the Broadwell Microarchitecture

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		31:8		Reserved
		41:32		RMID (R/W)
		63:42		Reserved
C8FH	3215	IA32_PQR_ASSOC	THREAD	Resource Association Register (R/W)
		9:0		RMID
		31:10		Reserved
		51:32		COS (R/W)
		63: 52		Reserved
C90H	3216	IA32_L3_QOS_MASK_0	Package	L3 Class Of Service Mask - COS 0 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=0.
		0:19		CBM: Bit vector of available L3 ways for COS 0 enforcement.
		63:20		Reserved
C91H	3217	IA32_L3_QOS_MASK_1	Package	L3 Class Of Service Mask - COS 1 (R/W)
				If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=1.
		0:19		CBM: Bit vector of available L3 ways for COS 1 enforcement.
		63:20		Reserved
C92H	3218	IA32_L3_QOS_MASK_2	Package	L3 Class Of Service Mask - COS 2 (R/W)
				If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=2.
		0:19		CBM: Bit vector of available L3 ways for COS 2 enforcement.
		63:20		Reserved
C93H	3219	IA32_L3_QOS_MASK_3	Package	L3 Class Of Service Mask - COS 3 (R/W)
				If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=3.
		0:19		CBM: Bit vector of available L3 ways for COS 3 enforcement.
		63:20		Reserved
C94H	3220	IA32_L3_QOS_MASK_4	Package	L3 Class Of Service Mask - COS 4 (R/W)
				If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=4.
		0:19		CBM: Bit vector of available L3 ways for COS 4 enforcement.
		63:20		Reserved
C95H	3221	IA32_L3_QOS_MASK_5	Package	L3 Class Of Service Mask - COS 5 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=5.
		0:19		CBM: Bit vector of available L3 ways for COS 5 enforcement.
		63:20		Reserved

Table 2-36. Additional MSRs Common to Intel® Xeon® Processor D and Intel Xeon Processors E5 v4 Family Based on the Broadwell Microarchitecture

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
C96H	3222	IA32_L3_QOS_MASK_6	Package	L3 Class Of Service Mask - COS 6 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=6.
		0:19		CBM: Bit vector of available L3 ways for COS 6 enforcement.
		63:20		Reserved
C97H	3223	IA32_L3_QOS_MASK_7	Package	L3 Class Of Service Mask - COS 7 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=7.
		0:19		CBM: Bit vector of available L3 ways for COS 7 enforcement.
		63:20		Reserved
C98H	3224	IA32_L3_QOS_MASK_8	Package	L3 Class Of Service Mask - COS 8 (R/W)
				If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=8.
		0:19		CBM: Bit vector of available L3 ways for COS 8 enforcement.
		63:20		Reserved
C99H	3225	IA32_L3_QOS_MASK_9	Package	L3 Class Of Service Mask - COS 9 (R/W)
				If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=9.
		0:19		CBM: Bit vector of available L3 ways for COS 9 enforcement.
		63:20		Reserved
C9AH	3226	IA32_L3_QOS_MASK_10	Package	L3 Class Of Service Mask - COS 10 (R/W)
				If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=10.
		0:19		CBM: Bit vector of available L3 ways for COS 10 enforcement.
		63:20		Reserved
C9BH	3227	IA32_L3_QOS_MASK_11	Package	L3 Class Of Service Mask - COS 11 (R/W)
				If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=11.
		0:19		CBM: Bit vector of available L3 ways for COS 11 enforcement.
		63:20		Reserved
C9CH	3228	IA32_L3_QOS_MASK_12	Package	L3 Class Of Service Mask - COS 12 (R/W)
				If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=12.
		0:19		CBM: Bit vector of available L3 ways for COS 12 enforcement.
		63:20		Reserved
C9DH	3229	IA32_L3_QOS_MASK_13	Package	L3 Class Of Service Mask - COS 13 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=13.
		0:19		CBM: Bit vector of available L3 ways for COS 13 enforcement.
		63:20		Reserved

Table 2-36. Additional MSRs Common to Intel® Xeon® Processor D and Intel Xeon Processors E5 v4 Family Based on the Broadwell Microarchitecture

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
C9EH	3230	IA32_L3_QOS_MASK_14	Package	L3 Class Of Service Mask - COS 14 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=14.
		0:19		CBM: Bit vector of available L3 ways for COS 14 enforcement.
		63:20		Reserved
C9FH	3231	IA32_L3_QOS_MASK_15	Package	L3 Class Of Service Mask - COS 15 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=15.
		0:19		CBM: Bit vector of available L3 ways for COS 15 enforcement.
		63:20		Reserved

2.16.1 Additional MSRs Supported in the Intel® Xeon® Processor D Product Family

The MSRs listed in Table 2-37 are available to Intel[®] Xeon[®] Processor D Product Family (CPUID DisplayFamily_DisplayModel = 06_56H). The Intel[®] Xeon[®] processor D product family is based on the Broadwell microarchitecture and supports the MSR interfaces listed in Table 2-20, Table 2-29, Table 2-34, Table 2-36, and Table 2-37.

Table 2-37. Additional MSRs Supported by Intel® Xeon® Processor D with DisplayFamily_DisplayModel 06_56H

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
1ACH	428	MSR_TURBO_RATIO_LIMIT3	Package	Config Ratio Limit of Turbo Mode R/O if MSR_PLATFORM_INFO.[28] = 0. R/W if MSR_PLATFORM_INFO.[28] = 1.
		62:0	Package	Reserved
		63	Package	Semaphore for Turbo Ratio Limit Configuration If 1, the processor uses override configuration specified in MSR_TURBO_RATIO_LIMIT, MSR_TURBO_RATIO_LIMIT1. If 0, the processor uses factory-set configuration (Default).
286H	646	IA32_MC6_CTL2	Package	See Table 2-2.
287H	647	IA32_MC7_CTL2	Package	See Table 2-2.
289H	649	IA32_MC9_CTL2	Package	See Table 2-2.
28AH	650	IA32_MC10_CTL2	Package	See Table 2-2.
291H	657	IA32_MC17_CTL2	Package	See Table 2-2.
292H	658	IA32_MC18_CTL2	Package	See Table 2-2.
293H	659	IA32_MC19_CTL2	Package	See Table 2-2.

Table 2-37. Additional MSRs Supported by Intel® Xeon® Processor D with DisplayFamily_DisplayModel 06_56H

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
418H	1048	IA32_MC6_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
419H	1049	IA32_MC6_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
41AH	1050	IA32_MC6_ADDR	Package	Bank MC6 reports MC errors from the integrated I/O module.
41BH	1051	IA32_MC6_MISC	Package	
41CH	1052	IA32_MC7_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
41DH	1053	IA32_MC7_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
41EH	1054	IA32_MC7_ADDR	Package	Bank MC7 reports MC errors from the home agent HA 0.
41FH	1055	IA32_MC7_MISC	Package	
424H	1060	IA32_MC9_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
425H	1061	IA32_MC9_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
426H	1062	IA32_MC9_ADDR	Package	Banks MC9 through MC 10 report MC errors from each channel of the integrated memory controllers.
427H	1063	IA32_MC9_MISC	Package	
428H	1064	IA32_MC10_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
429H	1065	IA32_MC10_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
42AH	1066	IA32_MC10_ADDR	Package	Banks MC9 through MC 10 report MC errors from each channel of the integrated memory controllers.
42BH	1067	IA32_MC10_MISC	Package	
444H	1092	IA32_MC17_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
445H	1093	IA32_MC17_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
446H	1094	IA32_MC17_ADDR	Package	Bank MC17 reports MC errors from the following pair of CBo/L3 Slices (if the pair is present): CBo0, CBo3, CBo6,
447H	1095	IA32_MC17_MISC	Package	CBo9, CBo12, CBo15.
448H	1096	IA32_MC18_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
449H	1097	IA32_MC18_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
44AH	1098	IA32_MC18_ADDR	Package	Bank MC18 reports MC errors from the following pair of CBo/L3 Slices (if the pair is present): CBo1, CBo4, CBo7,
44BH	1099	IA32_MC18_MISC	Package	CBo10, CBo13, CBo16.
44CH	1100	IA32_MC19_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
44DH	1101	IA32_MC19_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
44EH	1102	IA32_MC19_ADDR	Package	Bank MC19 reports MC errors from the following pair of CBo/L3 Slices (if the pair is present): CBo2, CBo5, CBo8 CBo11, CBo14, CBo17.
44FH	1103	IA32_MC19_MISC	Package	

See Table 2-20, Table 2-29, Table 2-34, and Table 2-36 for other MSR definitions applicable to processors with CPUID signature 06_56H.

NOTES:

2.16.2 Additional MSRs Supported in Intel® Xeon® Processors E5 v4 and E7 v4 Families

The MSRs listed in Table 2-37 are available to $Intel^{(8)}$ Xeon⁽⁸⁾ Processor E5 v4 and E7 v4 Families (CPUID DisplayFamily_DisplayModel = 06_4FH). The $Intel^{(8)}$ Xeon⁽⁸⁾ processor E5 v4 family is based on the Broadwell micro-

^{1.} An override configuration lower than the factory-set configuration is always supported. An override configuration higher than the factory-set configuration is dependent on features specific to the processor and the platform.

architecture and supports the MSR interfaces listed in Table 2-20, Table 2-21, Table 2-29, Table 2-34, Table 2-36, and Table 2-38.

Table 2-38. Additional MSRs Supported by Intel® Xeon® Processors with DisplayFamily_DisplayModel 06_4FH

Reg	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
1ACH	428	MSR_TURBO_RATIO_LIMIT3	Package	Config Ratio Limit of Turbo Mode R/O if MSR_PLATFORM_INFO.[28] = 0. R/W if MSR_PLATFORM_INFO.[28] = 1.
		62:0	Package	Reserved
		63	Package	Semaphore for Turbo Ratio Limit Configuration If 1, the processor uses override configuration specified in MSR_TURBO_RATIO_LIMIT, MSR_TURBO_RATIO_LIMIT1, and MSR_TURBO_RATIO_LIMIT2.
				If 0, the processor uses factory-set configuration (Default).
285H	645	IA32_MC5_CTL2	Package	See Table 2-2.
286H	646	IA32_MC6_CTL2	Package	See Table 2-2.
287H	647	IA32_MC7_CTL2	Package	See Table 2-2.
288H	648	IA32_MC8_CTL2	Package	See Table 2-2.
289H	649	IA32_MC9_CTL2	Package	See Table 2-2.
28AH	650	IA32_MC10_CTL2	Package	See Table 2-2.
28BH	651	IA32_MC11_CTL2	Package	See Table 2-2.
28CH	652	IA32_MC12_CTL2	Package	See Table 2-2.
28DH	653	IA32_MC13_CTL2	Package	See Table 2-2.
28EH	654	IA32_MC14_CTL2	Package	See Table 2-2.
28FH	655	IA32_MC15_CTL2	Package	See Table 2-2.
290H	656	IA32_MC16_CTL2	Package	See Table 2-2.
291H	657	IA32_MC17_CTL2	Package	See Table 2-2.
292H	658	IA32_MC18_CTL2	Package	See Table 2-2.
293H	659	IA32_MC19_CTL2	Package	See Table 2-2.
294H	660	IA32_MC20_CTL2	Package	See Table 2-2.
295H	661	IA32_MC21_CTL2	Package	See Table 2-2.
414H	1044	IA32_MC5_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
415H	1045	IA32_MC5_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.". Bank MC5 reports MC errors from the Intel QPI 0
416H	1046	IA32_MC5_ADDR	Package	module.
417H	1047	IA32_MC5_MISC	Package	
418H	1048	IA32_MC6_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
419H	1049	IA32_MC6_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.". Bank MC6 reports MC errors from the integrated I/O
41AH	1050	IA32_MC6_ADDR	Package	module.
41BH	1051	IA32_MC6_MISC	Package	

Table 2-38. Additional MSRs Supported by Intel® Xeon® Processors with DisplayFamily_DisplayModel 06_4FH

Hex Dec 41CH 1052 IA32_MC7_CTL Package Sec Section 16.3.2.1, "IA32_MCL_CTL MSRs" through 41CH 1053 IA32_MC7_STATUS Package Section 16.3.2.4, "IA32_MCL_MISC MSRs". 41CH 1054 IA32_MC7_MISC Package Section 16.3.2.4, "IA32_MCL_CTL MSRs" through 42CH 1056 IA32_MCB_CTL Package Section 16.3.2.1, "IA32_MCL_CTL MSRs" through 42CH 1058 IA32_MCB_ADDR Package Section 16.3.2.1, "IA32_MCL_CTL MSRs" through 42CH 1058 IA32_MCB_STATUS Package Sec Section 16.3.2.1, "IA32_MCL_CTL MSRs" through 42CH 1060 IA32_MCB_STATUS Package Sec Section 16.3.2.1, "IA32_MCL_CTL MSRs" through 42CH 1063 IA32_MCB_MCB_ADDR Package Sec Section 16.3.2.1, "IA32_MCL_CTL MSRs" through 42CH 1063 IA32_MCI_O_STATUS Package Sec Section 16.3.2.1, "IA32_MCL_CTL MSRs" through 42CH 1063 IA32_MCI_O_STATUS Package Sec Section 16.3.2.1, "IA32_MCL_CTL MSRs" through 42CH 1066 IA32_MCI_O_STATUS Package Sec	Register Address		Register Name / Bit Fields	Scope	Bit Description
A1DH 1053 IA32_MC7_STATUS Package Section 16.3.2.4, "IA32_MC_MISC MSRs." Bank MC7 reports MC errors from the home agent HA	Hex	Dec			
High 1054 A32_MC7_AIDR Package A1FH 1055 A32_MC7_MISC Package A2FH 1056 A32_MC8_STATUS Package See Section 16.3.2.1, "A32_MC_MISC MSRs." See Section 16.3.2.1, "A32_MC_MISC MSRs." Package See Section 16.3.2.1, "A32_MC_MISC MSRs." P	41CH	1052	IA32_MC7_CTL	Package	
41EH 1054 IA32_MC7_AUDK Package 0. 42PH 1055 IA32_MC8_CTL Package Sec Section 16.3.2.1, "IA32_MCL_MISC MSRs." 42PH 1057 IA32_MCB_STATUS Package Section 16.3.2.1, "IA32_MCL_MISC MSRs." 42PH 1058 IA32_MCB_ADDR Package Section 16.3.2.1, "IA32_MCL_MISC MSRs." 42PH 1060 IA32_MC9_STATUS Package See Section 16.3.2.1, "IA32_MCL_MISC MSRs." 42BH 1061 IA32_MC9_STATUS Package Section 16.3.2.4, "IA32_MCL_MISC MSRs." 42BH 1062 IA32_MC9_MISC Package Banks MG9 through MC 16 report MC errors from each channel of the integrated memory controllers. 42BH 1064 IA32_MC10_STATUS Package See Section 16.3.2.1, "IA32_MCL_MISC MSRs." 42BH 1065 IA32_MC10_MSC Package See Section 16.3.2.1, "IA32_MCL_MISC MSRs." 42BH 1067 IA32_MC10_MSC Package See Section 16.3.2.1, "IA32_MCL_MISC MSRs." 42BH 1067 IA32_MC10_MSC Package See Section 16.3.2.1, "IA32_MCL_MISC MSRs." 42BH 1067	41DH	1053	IA32_MC7_STATUS	Package	Bank MC7 reports MC errors from the home agent HA
420H 1056 IA32_MCB_CTL Package See Section 16.3.2.1, "IA32_MCI_CTL MSRs" through 421H 1057 IA32_MCB_STATUS Package Section 16.3.2.1, "IA32_MCI_MISC MSRs.". 422H 1058 IA32_MCB_ADDR Package Bank MC8 reports MC errors from the home agent HA 423H 1059 IA32_MCB_MSC Package See Section 16.3.2.1, "IA32_MCI_CTL MSRs" through 425H 1061 IA32_MC9_ADDR Package See Section 16.3.2.4, "IA32_MCI_MISC MSRs.". 426H 1062 IA32_MC9_ADDR Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs.". 427H 1063 IA32_MC10_CTL Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs.". 428H 1064 IA32_MC10_CTL Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs." 429H 1065 IA32_MC10_CTL Package See Section 16.3.2.1, "IA32_MCI_CTL MSRs" through 420H 1066 IA32_MC10_CTL Package See Section 16.3.2.1, "IA32_MCI_CTL MSRs" through 420H 1069 IA32_MC11_CTL Package See Section 16.3.2.1, "IA32_MCI_CTL MSRs" through 420H <td>41EH</td> <td>1054</td> <td>IA32_MC7_ADDR</td> <td>Package</td>	41EH	1054	IA32_MC7_ADDR	Package	
421H 1057 IA32_MCB_STATUS Package Section 16.3.2.4, "IA32_MG_MISC MSRs." 422H 1058 IA32_MCB_ADDR Package 1. 423H 1059 IA32_MCB_MISC Package 1. 424H 1060 IA32_MC9_CTL Package Section 16.3.2.1, "IA32_MCI_MISC MSRs.". 425H 1061 IA32_MC9_ADDR Package Section 16.3.2.4, "IA32_MCI_MISC MSRs.". 426H 1062 IA32_MC9_MISC Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs.". 428H 1064 IA32_MC10_CTL Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs.". 428H 1064 IA32_MC10_STATUS Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs.". 428H 1066 IA32_MC10_DISC Package Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers. 42BH 1066 IA32_MC1_ADDR Package See Section 16.3.2.1, "IA32_MCI_CTL MSRs" through MSS_MC1 MSS_M	41FH	1055	IA32_MC7_MISC	Package	
422H 1059 IA32_MC8_ADDR Package 423H 1059 IA32_MC8_ADDR Package 424H 1060 IA32_MC9_CTL Package 425H 1061 IA32_MC9_CTL Package 426H 1062 IA32_MC9_ADDR Package 427H 1063 IA32_MC9_ADDR Package 428H 1064 IA32_MC10_GTL Package 428H 1064 IA32_MC10_GTL Package 429H 1065 IA32_MC10_STATUS Package 429H 1066 IA32_MC10_ADDR Package 429H 1065 IA32_MC10_ADDR Package 420H 1068 IA32_MC11_GTL Package 420H 1069 IA32_MC11_STATUS Package 420H 1069 IA32_MC11_STATUS Package 420H 1070 IA32_MC11_STATUS Package 430H 1072 IA32_MC12_GTL Package 430H 1073 IA32_MC13_ADDR Package	420H	1056	IA32_MC8_CTL	Package	
423H 1059 IA32_MCB_MISC Package 1. 424H 1060 IA32_MC9_CTL Package See Section 16.3.2.1, "IA32_MCI_CTL MSRs" through 425H 1061 IA32_MC9_ADDR Package Section 16.3.2.4, "IA32_MCI_MISC MSRs"." 426H 1062 IA32_MC9_MISC Package Section 16.3.2.1, "IA32_MCI_CTL MSRs" through 428H 1064 IA32_MC10_CTL Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs"." 429H 1065 IA32_MC10_ADDR Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs"." 428H 1067 IA32_MC10_MISC Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs"." 42BH 1067 IA32_MC11_CTL Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs"." 42CH 1068 IA32_MC11_MISC Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs"." 42CH 1070 IA32_MC11_MISC Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs"." 43DH 1072 IA32_MC12_STATUS Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs"." 432H 1074 IA32_MC13_MISC	421H	1057	IA32_MC8_STATUS	Package	
423H 1059 IA32_MC8_MISC Package 424H 1060 IA32_MC9_CTL Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs." 426H 1061 IA32_MC9_STATUS Package Section 16.3.2.4, "IA32_MCI_MISC MSRs." 426H 1062 IA32_MC9_MISC Package Banks MC9 through MC16 report MC errors from each channel of the integrated memory controllers. 428H 1064 IA32_MC10_STATUS Package See Section 16.3.2.1, "IA32_MCI_CTL MSRs" through Section 16.3.2.4, "IA32_MCI_CTL MSRs". 428H 1066 IA32_MC10_ADDR Package Section 16.3.2.4, "IA32_MCI_CTL MSRs" through Section 16.3.2.4, "IA32_MCI_MISC MSRs". 420H 1069 IA32_MC11_STATUS Package See Section 16.3.2.1, "IA32_MCI_CTL MSRs" through Section 16.3.2.4, "IA32_MCI_MISC MSRs". 430H 1074 IA32_MC13_STATUS Package Section 16.3.2.4, "IA32_MCI_MISC MSRs". 438H 1076 IA32_MC13_MDR Package See Section 16.3.2.4, "IA32_MCI_MI	422H	1058	IA32_MC8_ADDR	Package	
425H 1061 IA32_MC9_STATUS Package 426H 1062 IA32_MC9_ADDR Package 427H 1063 IA32_MC9_MISC Package 428H 1064 IA32_MC10_CTL Package 428H 1065 IA32_MC10_STATUS Package 428H 1066 IA32_MC10_ADDR Package 42BH 1067 IA32_MC10_MISC Package 42BH 1067 IA32_MC11_STATUS Package 42DH 1068 IA32_MC11_STATUS Package 42DH 1069 IA32_MC11_STATUS Package 42DH 1069 IA32_MC11_STATUS Package 42DH 1069 IA32_MC11_STATUS Package 42DH 1070 IA32_MC11_STATUS Package 42FH 1071 IA32_MC12_CTL Package 430H 1072 IA32_MC12_STATUS Package 431H 1073 IA32_MC12_MISC Package 432H 1074 IA32_MC13_STATUS Package	423H	1059	IA32_MC8_MISC	Package	
A25H 1061 A32_MC19_ADDR	424H	1060	IA32_MC9_CTL	Package	
426H1062IA32_MC19_MISCPackage428H1064IA32_MC10_CTLPackage429H1065IA32_MC10_STATUSPackage428H1066IA32_MC10_ADDRPackage428H1067IA32_MC10_MISCPackage428H1067IA32_MC10_MISCPackage42BH1067IA32_MC11_STATUSPackage42CH1068IA32_MC11_STATUSPackage42EH1070IA32_MC11_STATUSPackage42EH1070IA32_MC11_MISCPackage42EH1071IA32_MC11_MISCPackage43BH1072IA32_MC12_STATUSPackage43H1073IA32_MC12_STATUSPackage43H1074IA32_MC12_STATUSPackage43H1075IA32_MC12_MISCPackage43H1076IA32_MC12_MISCPackage43H1076IA32_MC13_STATUSPackage43H1076IA32_MC13_STATUSPackage43H1079IA32_MC13_STATUSPackage43H1079IA32_MC13_STATUSPackage43H1079IA32_MC13_STATUSPackage43H1079IA32_MC13_STATUSPackage43H1079IA32_MC13_STATUSPackage43H1079IA32_MC14_CTLPackage43H1079IA32_MC14_CTLPackage43H1080IA32_MC14_STATUSPackage43H1081IA32_MC14_STATUSPackage43H1081 </td <td>425H</td> <td>1061</td> <td>IA32_MC9_STATUS</td> <td>Package</td> <td></td>	425H	1061	IA32_MC9_STATUS	Package	
427H 1063 IA32_MC19_MISC Package 428H 1064 IA32_MC10_CTL Package See Section 16.3.2.1, "IA32_MCI_CTL MSRs" through 429H 1065 IA32_MC10_STATUS Package Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers. 42BH 1067 IA32_MC10_MISC Package See Section 16.3.2.1, "IA32_MCI_CTL MSRs" through Section 16.3.2.4, "IA32_MCI_CTL MSRs" through Section 16.3.2.4, "IA32_MCI_CTL MSRs" through Section 16.3.2.4, "IA32_MCI_MISC MSRs.". 42DH 1069 IA32_MC11_ADDR Package See Section 16.3.2.1, "IA32_MCI_CTL MSRs" through Section 16.3.2.4, "IA32_MCI_MISC MSRs.". 43H 1072 IA32_MC12_ADDR Package See Section 16.3.2.1, "IA32_MCI_CTL MSRs" through Section 16.3.2.4, "IA32_MCI_CTL MSRs" through S	426H	1062	IA32_MC9_ADDR	Package	
429H 1065 IA32_MC10_STATUS Package 42AH 1066 IA32_MC10_ADDR Package 42BH 1067 IA32_MC10_MISC Package 42CH 1068 IA32_MC11_CTL Package 42DH 1069 IA32_MC11_STATUS Package 42EH 1070 IA32_MC11_ADDR Package 42FH 1071 IA32_MC11_ADDR Package 42FH 1071 IA32_MC11_MISC Package 430H 1072 IA32_MC12_CTL Package 431H 1073 IA32_MC12_STATUS Package 431H 1074 IA32_MC12_MISC Package 43H 1075 IA32_MC13_ADDR Package 43H 1076 IA32_MC13_CTL Package 43H 1076 IA32_MC13_STATUS Package 43H 1076 IA32_MC13_ADDR Package 43H 1076 IA32_MC13_ADDR Package 43H 1079 IA32_MC13_MISC Package <	427H	1063	IA32_MC9_MISC	Package	
42AH 1066 IA32_MC10_ADDR Package 42BH 1067 IA32_MC10_ADDR Package 42CH 1068 IA32_MC10_MISC Package 42CH 1068 IA32_MC11_CTL Package 42CH 1069 IA32_MC11_STATUS Package 42CH 1070 IA32_MC11_ADDR Package 42CH 1071 IA32_MC11_ADDR Package 42CH 1071 IA32_MC11_ADDR Package 43DH 1072 IA32_MC12_CTL Package 43DH 1073 IA32_MC12_STATUS Package 43H 1074 IA32_MC12_ADDR Package 43H 1075 IA32_MC12_ADDR Package 43H 1076 IA32_MC13_ADDR Package 43H 1077 IA32_MC13_STATUS Package 43H 1079 IA32_MC13_STATUS Package 43H 1070 IA32_MC13_STATUS Package 43H 108D IA32_MC14_CTL Package 43H 108D IA32_MC14_STATUS Package 43H 108D IA32_MC15_STATUS Package	428H	1064	IA32_MC10_CTL	Package	
42AH 1066 IA32_MC10_MISC Package channel of the integrated memory controllers. 42BH 1067 IA32_MC11_CTL Package See Section 16.3.2.1, "IA32_MCI_CTL MSRs" through 42DH 1069 IA32_MC11_STATUS Package Section 16.3.2.1, "IA32_MCI_MISC MSRs.". 42EH 1070 IA32_MC11_MISC Package Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers. 430H 1072 IA32_MC12_STATUS Package See Section 16.3.2.1, "IA32_MCI_CTL MSRs" through Section 16.3.2.4, "IA32_MCI_MISC MSRs.". 431H 1073 IA32_MC12_ADDR Package Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers. 433H 1075 IA32_MC13_MISC Package Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers. 435H 1076 IA32_MC13_STATUS Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs.". 438H 1079 IA32_MC13_ADDR Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs.". 438H 1080 IA32_MC14_CTL Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs.".	429H	1065	IA32_MC10_STATUS	Package	
42BH 1067 IA32_MC10_MISC Package See Section 16.3.2.1, "IA32_MCI_CTL MSRs" through 42DH 1069 IA32_MC11_STATUS Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs.". 42EH 1070 IA32_MC11_ADDR Package Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers. 42FH 1071 IA32_MC12_CTL Package See Section 16.3.2.1, "IA32_MCI_CTL MSRs" through Section 16.3.2.4, "IA32_MCI_CTL MSRs" through Section 16.3.2.4, "IA32_MCI_MISC MSRs.". 43H 1073 IA32_MC12_ADDR Package Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers. 43H 1074 IA32_MC12_MISC Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs.". 43H 1076 IA32_MC13_STATUS Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs.". 43H 1077 IA32_MC13_ADDR Package Section 16.3.2.1, "IA32_MCI_MISC MSRs.". 43H 1079 IA32_MC14_CTL Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs.". 43H 1080 IA32_MC14_STATUS Package See Section 16.3.2.1, "IA32_MCI_MISC MSRs.". 43H	42AH	1066	IA32_MC10_ADDR	Package	
42DH 1069 IA32_MC11_STATUS Package 42EH 1070 IA32_MC11_ADDR Package 42FH 1071 IA32_MC11_MISC Package 430H 1072 IA32_MC12_CTL Package 431H 1073 IA32_MC12_STATUS Package 432H 1074 IA32_MC12_ADDR Package 433H 1075 IA32_MC13_MISC Package 434H 1076 IA32_MC13_CTL Package 435H 1077 IA32_MC13_STATUS Package 436H 1078 IA32_MC13_ADDR Package 437H 1079 IA32_MC13_MISC Package 438H 1080 IA32_MC14_STATUS Package 439H 1081 IA32_MC14_STATUS Package 438H 1080 IA32_MC14_MISC Package 438H 1080 IA32_MC14_STATUS Package 438H 1081 IA32_MC14_STATUS Package 438H 1082 IA32_MC14_MISC Package	42BH	1067	IA32_MC10_MISC	Package	
42EH 1070 IA32_MC11_ADDR Package 430H 1072 IA32_MC11_MISC 430H 1073 IA32_MC12_CTL 431H 1073 IA32_MC12_STATUS 432H 1074 IA32_MC12_ADDR 433H 1075 IA32_MC12_MISC 434H 1076 IA32_MC13_CTL 435H 1077 IA32_MC13_CTL 436H 1078 IA32_MC13_STATUS 437H 1079 IA32_MC13_MISC 438H 1080 IA32_MC13_MISC 438H 1080 IA32_MC14_CTL 438H 1080 IA32_MC14_CTL 438H 1081 IA32_MC14_STATUS 438H 1082 IA32_MC14_STATUS 438H 1082 IA32_MC14_STATUS 438H 1082 IA32_MC14_MISC 438H 1084 IA32_MC15_CTL 438H 1085 IA32_MC15_CTL 438H 1086 IA32_MC15_CTL 438H 1086 IA32_MC15_STATUS 438H 1088 IA32_MC15_CTL 438H 1080 IA32_MC16_STATUS 438H 1080 IA32_MC16_CTL 438H 1080 IA32_MC16_CTL 438H 1080 IA32_MC16_STATUS 438H 1080 IA32_MC16_STATUS 438H 1080 IA32_MC16_CTL 438H	42CH	1068	IA32_MC11_CTL	Package	
42EH1070IA32_MC11_ADDRPackagechannel of the integrated memory controllers.42FH1071IA32_MC11_MISCPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through430H1072IA32_MC12_STATUSPackageSee Section 16.3.2.1, "IA32_MCi_MISC MSRs.".431H1073IA32_MC12_ADDRPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.433H1075IA32_MC13_MISCPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_MISC MSRs.".435H1077IA32_MC13_STATUSPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.436H1078IA32_MC13_ADDRPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.437H1079IA32_MC14_CTLPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_MISC MSRs.".438H1080IA32_MC14_STATUSPackageSection 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_MISC MSRs.".43BH1081IA32_MC14_ADDRPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.43CH1084IA32_MC15_CTLPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_MISC MSRs.".43CH1086IA32_MC15_STATUSPackageSee Section 16.3.2.4, "IA32_MCi_CTL MSRs" through MC 16 report MC errors from each channel of the integrated memory contro	42DH	1069	IA32_MC11_STATUS	Package	
42FH 1071 IA32_MC11_MISC Package 430H 1072 IA32_MC12_CTL Package See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_MISC MSRs.". 431H 1073 IA32_MC12_STATUS Package Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers. 433H 1075 IA32_MC12_MISC Package See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_MISC MSRs.". 435H 1077 IA32_MC13_STATUS Package See Section 16.3.2.1, "IA32_MCi_MISC MSRs.". 436H 1078 IA32_MC13_ADDR Package Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers. 437H 1079 IA32_MC13_MISC Package See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.1, "IA32_MCi_MISC MSRs.". 438H 1080 IA32_MC14_STATUS Package See Section 16.3.2.1, "IA32_MCi_MISC MSRs.". 43AH 1082 IA32_MC14_ADDR Package Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers. 43DH 1085 IA32_MC15_STATUS Package <td< td=""><td>42EH</td><td>1070</td><td>IA32_MC11_ADDR</td><td>Package</td><td></td></td<>	42EH	1070	IA32_MC11_ADDR	Package	
431H1073IA32_MC12_STATUSPackage432H1074IA32_MC12_ADDRPackage433H1075IA32_MC12_MISCPackage434H1076IA32_MC13_CTLPackage435H1077IA32_MC13_STATUSPackage436H1078IA32_MC13_ADDRPackage437H1079IA32_MC13_MISCPackage438H1080IA32_MC14_CTLPackage439H1081IA32_MC14_STATUSPackage438H1082IA32_MC14_STATUSPackage438H1082IA32_MC14_STATUSPackage438H1083IA32_MC14_ADDRPackage438H1084IA32_MC14_MISCPackage438H1085IA32_MC15_CTLPackage43CH1084IA32_MC15_CTLPackage43CH1084IA32_MC15_STATUSPackage43CH1085IA32_MC15_STATUSPackage43CH1086IA32_MC15_STATUSPackage43CH1086IA32_MC15_ADDRPackage43CH1086IA32_MC15_ADDRPackage43CH1086IA32_MC15_ADDRPackage43CH1086IA32_MC15_ADDRPackage43CH1086IA32_MC15_ADDRPackage	42FH	1071	IA32_MC11_MISC	Package	
431H1073IA32_MC12_STATUSPackage432H1074IA32_MC12_ADDRPackage433H1075IA32_MC12_MISCPackage434H1076IA32_MC13_CTLPackage435H1077IA32_MC13_STATUSPackage436H1078IA32_MC13_ADDRPackage437H1079IA32_MC13_ADDRPackage438H1080IA32_MC13_MISCPackage439H1081IA32_MC14_STATUSPackage43AH1082IA32_MC14_STATUSPackage43BH1083IA32_MC14_ADDRPackage43CH1084IA32_MC15_CTLPackage43DH1085IA32_MC15_STATUSPackage43CH1086IA32_MC15_STATUSPackage43CH1086IA32_MC15_STATUSPackage43CH1086IA32_MC15_STATUSPackage43CH1086IA32_MC15_STATUSPackage43CH1086IA32_MC15_STATUSPackage43CH1086IA32_MC15_STATUSPackage43CH1086IA32_MC15_STATUSPackage43CH1086IA32_MC15_ADDRPackage	430H	1072	IA32_MC12_CTL	Package	
432H1074IA32_MC12_ADDRPackagechannel of the integrated memory controllers.433H1075IA32_MC13_CTLPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through434H1076IA32_MC13_CTLPackageSection 16.3.2.4, "IA32_MCi_MISC MSRs.".435H1077IA32_MC13_ADDRPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.437H1079IA32_MC13_MISCPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through438H1080IA32_MC14_CTLPackageSee Section 16.3.2.4, "IA32_MCi_CTL MSRs" through439H1081IA32_MC14_STATUSPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.43AH1082IA32_MC14_ADDRPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through43BH1083IA32_MC14_MISCPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through43DH1085IA32_MC15_CTLPackageSee Section 16.3.2.4, "IA32_MCi_MISC MSRs.".43CH1086IA32_MC15_STATUSPackageSection 16.3.2.4, "IA32_MCi_MISC MSRs.".43CH1086IA32_MC15_ADDRPackageSend through MC 16 report MC errors from each channel of the integrated memory controllers.	431H	1073	IA32_MC12_STATUS	Package	· — — —
433H1075IA32_MC12_MISCPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through434H1076IA32_MC13_STATUSPackageSee Section 16.3.2.1, "IA32_MCi_MISC MSRs.".435H1077IA32_MC13_STATUSPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.437H1079IA32_MC13_MISCPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_MISC MSRs.".438H1080IA32_MC14_CTLPackageSee Section 16.3.2.4, "IA32_MCi_MISC MSRs.".439H1081IA32_MC14_STATUSPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.43BH1082IA32_MC14_MISCPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_MISC MSRs.".43DH1085IA32_MC15_CTLPackageSee Section 16.3.2.1, "IA32_MCi_MISC MSRs.".43CH1086IA32_MC15_ADDRPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.	432H	1074	IA32_MC12_ADDR	Package	
435H1077IA32_MC13_STATUSPackageSection 16.3.2.4, "IA32_MCi_MISC MSRs.".436H1078IA32_MC13_ADDRPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.437H1079IA32_MC13_MISCPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_MISC MSRs.".439H1081IA32_MC14_STATUSPackageSee Section 16.3.2.4, "IA32_MCi_MISC MSRs.".43H1082IA32_MC14_ADDRPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.43H1083IA32_MC14_MISCPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_MISC MSRs.".43CH1085IA32_MC15_STATUSPackageSection 16.3.2.4, "IA32_MCi_MISC MSRs.".43CH1086IA32_MC15_ADDRPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.	433H	1075	IA32_MC12_MISC	Package	
435H1077IA32_MC13_STATUSPackage436H1078IA32_MC13_ADDRPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.437H1079IA32_MC13_MISCPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_MISC MSRs.".438H1081IA32_MC14_STATUSPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.43AH1082IA32_MC14_ADDRPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.43CH1084IA32_MC15_CTLPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_MISC MSRs.".43DH1085IA32_MC15_STATUSPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.	434H	1076	IA32_MC13_CTL	Package	
436H1078IA32_MC13_ADDRPackagechannel of the integrated memory controllers.437H1079IA32_MC13_MISCPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through438H1080IA32_MC14_CTLPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through439H1081IA32_MC14_STATUSPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.438H1082IA32_MC14_MISCPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through43CH1084IA32_MC15_CTLPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through43CH1085IA32_MC15_STATUSPackageSection 16.3.2.4, "IA32_MCi_MISC MSRs.".43CH1086IA32_MC15_ADDRPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.	435H	1077	IA32_MC13_STATUS	Package	
438H 1080 IA32_MC14_CTL Package See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_MISC MSRs.". 439H 1081 IA32_MC14_STATUS Package Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers. 438H 1083 IA32_MC14_MISC Package See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through MC 16 report MC errors from each channel of the integrated memory controllers. 43CH 1084 IA32_MC15_CTL Package See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_MISC MSRs.". 43CH 1086 IA32_MC15_STATUS Package Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.	436H	1078	IA32_MC13_ADDR	Package	
439H1081IA32_MC14_STATUSPackageSection 16.3.2.4, "IA32_MCi_MISC MSRs.".43AH1082IA32_MC14_ADDRPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.43BH1083IA32_MC14_MISCPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_MISC MSRs.".43CH1085IA32_MC15_STATUSPackageSection 16.3.2.4, "IA32_MCi_MISC MSRs.".43CH1086IA32_MC15_ADDRPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.	437H	1079	IA32_MC13_MISC	Package	
439H1081IA32_MC14_STATUSPackage43AH1082IA32_MC14_ADDRPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.43BH1083IA32_MC14_MISCPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_MISC MSRs.".43DH1085IA32_MC15_STATUSPackageSection 16.3.2.4, "IA32_MCi_MISC MSRs.".43EH1086IA32_MC15_ADDRPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.	438H	1080	IA32_MC14_CTL	Package	
43AH1082IA32_MC14_ADDRPackagechannel of the integrated memory controllers.43BH1083IA32_MC14_MISCPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_MISC MSRs.".43DH1085IA32_MC15_STATUSPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.	439H	1081	IA32_MC14_STATUS	Package	
43BH1083IA32_MC14_MISCPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through43CH1084IA32_MC15_CTLPackageSee Section 16.3.2.1, "IA32_MCi_CTL MSRs" through Section 16.3.2.4, "IA32_MCi_MISC MSRs.".43CH1086IA32_MC15_ADDRPackageBanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.	43AH	1082	IA32_MC14_ADDR	Package	
43DH 1085 IA32_MC15_STATUS Package Section 16.3.2.4, "IA32_MCi_MISC MSRs.". 43EH 1086 IA32_MC15_ADDR Package Shanks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.	43BH	1083	IA32_MC14_MISC	Package	
43EH 1086 IA32_MC15_STATOS Package Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.	43CH	1084	IA32_MC15_CTL	Package	
43EH 1086 IA32_MC15_ADDR Package channel of the integrated memory controllers.	43DH	1085	IA32_MC15_STATUS	Package	
	43EH	1086	IA32_MC15_ADDR	Package	
	43FH	1087	IA32_MC15_MISC	Package	

Table 2-38. Additional MSRs Supported by Intel® Xeon® Processors with DisplayFamily_DisplayModel 06_4FH

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
440H	1088	IA32_MC16_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
441H	1089	IA32_MC16_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
442H	1090	IA32_MC16_ADDR	Package	Banks MC9 through MC 16 report MC errors from each channel of the integrated memory controllers.
443H	1091	IA32_MC16_MISC	Package	
444H	1092	IA32_MC17_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
445H	1093	IA32_MC17_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
446H	1094	IA32_MC17_ADDR	Package	Bank MC17 reports MC errors from the following pair of CBo/L3 Slices (if the pair is present): CBo0, CBo3, CBo6,
447H	1095	IA32_MC17_MISC	Package	CBo9, CBo12, CBo15.
448H	1096	IA32_MC18_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
449H	1097	IA32_MC18_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
44AH	1098	IA32_MC18_ADDR	Package	Bank MC18 reports MC errors from the following pair of CBo/L3 Slices (if the pair is present): CBo1, CBo4, CBo7,
44BH	1099	IA32_MC18_MISC	Package	CBo10, CBo13, CBo16.
44CH	1100	IA32_MC19_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
44DH	1101	IA32_MC19_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.". Bank MC19 reports MC errors from the following pair of
44EH	1102	IA32_MC19_ADDR	Package	CBo/L3 Slices (if the pair is present): CBo2, CBo5, CBo8,
44FH	1103	IA32_MC19_MISC	Package	CBo11, CBo14, CBo17.
450H	1104	IA32_MC20_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
451H	1105	IA32_MC20_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.". Bank MC20 reports MC errors from the Intel QPI 1
452H	1106	IA32_MC20_ADDR	Package	module.
453H	1107	IA32_MC20_MISC	Package	
454H	1108	IA32_MC21_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
455H	1109	IA32_MC21_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.". Bank MC21 reports MC errors from the Intel QPI 2
456H	1110	IA32_MC21_ADDR	Package	module.
457H	1111	IA32_MC21_MISC	Package	
C81H	3201	IA32_L3_QOS_CFG	Package	Cache Allocation Technology Configuration (R/W)
		0		CAT Enable. Set 1 to enable Cache Allocation Technology.
		63:1		Reserved

See Table 2-20, Table 2-21, Table 2-29, and Table 2-30 for other MSR definitions applicable to processors with CPUID signature 06_45H.

NOTES

1. An override configuration lower than the factory-set configuration is always supported. An override configuration higher than the factory-set configuration is dependent on features specific to the processor and the platform.

2.17 MSRS IN THE 6TH GENERATION, 7TH GENERATION, 8TH GENERATION, 9TH GENERATION, 10TH GENERATION, 11TH GENERATION, 12TH GENERATION, AND 13TH GENERATION INTEL® CORE™ PROCESSORS, INTEL® XEON® SCALABLE PROCESSOR FAMILY, 2ND, 3RD, AND 4TH GENERATION INTEL® XEON® SCALABLE PROCESSOR FAMILY, 8TH GENERATION INTEL® CORE™ I3 PROCESSORS, AND INTEL® XEON® E PROCESSORS

6th generation Intel[®] Core[™] processors are based on the Skylake microarchitecture and have CPUID DisplayFamily_DisplayModel signatures of 06_4EH and 06_5EH.

The Intel[®] Xeon[®] Scalable Processor Family based on the Skylake microarchitecture, the 2nd generation Intel[®] Xeon[®] Scalable Processor Family based on the Cascade Lake product, and the 3rd generation Intel[®] Xeon[®] Scalable Processor Family based on the Cooper Lake product all have a CPUID DisplayFamily_DisplayModel signature of 06_55H.

7th generation Intel[®] Core[™] processors are based on the Kaby Lake microarchitecture, 8th generation and 9th generation Intel[®] Core[™] processors and Intel[®] Xeon[®] E processors are based on the Coffee Lake microarchitecture; these processors have CPUID DisplayFamily_DisplayModel signatures of 06_8EH and 06_9EH.

8th generation Intel[®] Core[™] i3 processors are based on Cannon Lake microarchitecture and have a CPUID DisplayFamily DisplayModel signature of 06–66H.

10th generation Intel[®] Core[™] processors are based on Comet Lake microarchitecture (with CPUID DisplayFamily_DisplayModel signatures of 06_A5H, 06_A6H) and Ice Lake microarchitecture (with CPUID DisplayFamily DisplayModel signatures of 06_7DH and 06_7EH).

11th generation Intel[®] Core[™] processors are based on the Tiger Lake microarchitecture and have CPUID DisplayFamily_DisplayModel signatures of 06_8CH and 06_8DH.

The 3rd generation $Intel^{(R)}$ Xeon $^{(R)}$ Scalable Processor Family is based on Ice Lake microarchitecture and has CPUID DisplayFamily_DisplayModel signatures of 06_6AH and 06_6CH.

The 12th generation $Intel^{\otimes}$ $Core^{\text{TM}}$ processors supporting the Alder Lake performance hybrid architecture have CPUID DisplayFamily_DisplayModel signatures of 06_97H and 06_9AH.

The 13th generation Intel[®] Core[™] processors supporting the Raptor Lake performance hybrid architecture have CPUID DisplayFamily DisplayModel signatures of 06 BAH, 06 B7H, and 06 BFH.

The 4th generation Intel[®] Xeon[®] Scalable Processor Family is based on Sapphire Rapids microarchitecture and has a CPUID DisplayFamily_DisplayModel signature of 06_8FH.

These processors support the MSR interfaces listed in Table 2-20, Table 2-21, Table 2-25, Table 2-29, Table 2-35, and Table 2-39¹. For an MSR listed in Table 2-39 that also appears in the model-specific tables of prior generations, Table 2-39 supersede prior generation tables.

Tables 2-40 through 2-52 list additional supported MSR interfaces for specific processors; see each table for additional details.

The notation of "Platform" in the Scope column (with respect to MSR_PLATFORM_ENERGY_COUNTER and MSR_PLATFORM_POWER_LIMIT) is limited to the power-delivery domain and the specifics of the power delivery integration may vary by platform vendor's implementation.

^{1.} MSRs at the following addresses are not supported in the 12th generation Intel Core processor E-core: 3F7H. MSRs at the following addresses are not supported in the 12th generation Intel Core processor E-core or P-core: 652H, 653H, 655H, 656H, DB0H, DB1H, DB2H, and D90H.

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec]		
ЗАН	58	IA32_FEATURE_CONTROL	Thread	Control Features in Intel 64 Processor (R/W) See Table 2-2.
FEH	254	IA32_MTRRCAP	Thread	MTRR Capability (R/O, Architectural) See Table 2-2
19CH	412	IA32_THERM_STATUS	Core	Thermal Monitor Status (R/W) See Table 2-2.
		0		Thermal Status (R/O) See Table 2-2.
		1		Thermal Status Log (R/WCO) See Table 2-2.
		2		PROTCHOT # or FORCEPR# Status (R/O) See Table 2-2.
		3		PROTCHOT # or FORCEPR# Log (R/WCO) See Table 2-2.
		4		Critical Temperature Status (R/O) See Table 2-2.
		5		Critical Temperature Status Log (R/WC0) See Table 2-2.
		6		Thermal threshold #1 Status (R/O) See Table 2-2.
		7		Thermal threshold #1 Log (R/WC0) See Table 2-2.
		8		Thermal Threshold #2 Status (R/O) See Table 2-2.
		9		Thermal Threshold #2 Log (R/WC0) See Table 2-2.
		10		Power Limitation Status (R/O) See Table 2-2.
		11		Power Limitation Log (R/WC0) See Table 2-2.
		12		Current Limit Status (R/O) See Table 2-2.
		13		Current Limit Log (R/WCO) See Table 2-2.
		14		Cross Domain Limit Status (R/O) See Table 2-2.

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		15		Cross Domain Limit Log (R/WC0) See Table 2-2.
		22:16		Digital Readout (R/O) See Table 2-2.
		26:23		Reserved
		30:27		Resolution in Degrees Celsius (R/O) See Table 2-2.
		31		Reading Valid (R/O) See Table 2-2.
		63:32		Reserved
1ADH	429	MSR_TURBO_RATIO_LIMIT	Package	Maximum Ratio Limit of Turbo Mode R/O if MSR_PLATFORM_INFO.[28] = 0, R/W if MSR_PLATFORM_INFO.[28] = 1
		7:0	Package	Maximum Ratio Limit for 1C Maximum turbo ratio limit of 1 core active.
		15:8	Package	Maximum Ratio Limit for 2C Maximum turbo ratio limit of 2 core active.
		23:16	Package	Maximum Ratio Limit for 3C Maximum turbo ratio limit of 3 core active.
		31:24	Package	Maximum Ratio Limit for 4C Maximum turbo ratio limit of 4 core active.
		63:32		Reserved
1C9H	457	MSR_LASTBRANCH_TOS	Thread	Last Branch Record Stack TOS (R/W)
				Contains an index (bits 0-4) that points to the MSR containing the most recent branch record.
1FCH	508	MSR_POWER_CTL	Core	Power Control Register See http://biosbits.org.
		0		Reserved
		1	Package	C1E Enable (R/W)
				When set to '1', will enable the CPU to switch to the Minimum Enhanced Intel SpeedStep Technology operating point when all execution cores enter MWAIT (C1).
		18:2		Reserved

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		19		Disable Energy Efficiency Optimization (R/W) Setting this bit disables the P-States energy efficiency optimization. Default value is 0. Disable/enable the energy efficiency optimization in P-State legacy mode (when IA32_PM_ENABLE[HWP_ENABLE] = 0), has an effect only in the turbo range or into PERF_MIN_CTL value if it is not zero set. In HWP mode (IA32_PM_ENABLE[HWP_ENABLE] == 1), has an effect between the OS desired or OS maximize to the OS minimize performance setting.
		20		Disable Race to Halt Optimization (R/W) Setting this bit disables the Race to Halt optimization and avoids this optimization limitation to execute below the most efficient frequency ratio. Default value is 0 for processors that support Race to Halt optimization.
		63:21		Reserved
300H	768	MSR_SGXOWNEREPOCHO	Package	Lower 64 Bit CR_SGXOWNEREPOCH (W) Writes do not update CR_SGXOWNEREPOCH if CPUID.(EAX=12H, ECX=0):EAX.SGX1 is 1 on any thread in the package.
		63:0		Lower 64 bits of an 128-bit external entropy value for key derivation of an enclave.
301H	768	MSR_SGXOWNEREPOCH1	Package	Upper 64 Bit CR_SGXOWNEREPOCH (W) Writes do not update CR_SGXOWNEREPOCH if CPUID.(EAX=12H, ECX=0):EAX.SGX1 is 1 on any thread in the package.
		63:0		Upper 64 bits of an 128-bit external entropy value for key derivation of an enclave.
38EH	910	IA32_PERF_GLOBAL_STATUS		See Table 2-2. See Section 20.2.4, "Architectural Performance Monitoring Version 4."
		0	Thread	Ovf_PMC0
		1	Thread	Ovf_PMC1
		2	Thread	Ovf_PMC2
		3	Thread	Ovf_PMC3
		4	Thread	Ovf_PMC4 (if CPUID.OAH:EAX[15:8] > 4)
		5	Thread	Ovf_PMC5 (if CPUID.OAH:EAX[15:8] > 5)
		6	Thread	Ovf_PMC6 (if CPUID.OAH:EAX[15:8] > 6)
		7	Thread	Ovf_PMC7 (if CPUID.OAH:EAX[15:8] > 7)
		31:8		Reserved
		32	Thread	Ovf_FixedCtr0

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		33	Thread	Ovf_FixedCtr1
		34	Thread	Ovf_FixedCtr2
		54:35		Reserved
		55	Thread	Trace_ToPA_PMI
		57:56		Reserved
		58	Thread	LBR_Frz
		59	Thread	CTR_Frz
		60	Thread	ASCI
		61	Thread	Ovf_Uncore
		62	Thread	Ovf_BufDSSAVE
		63	Thread	CondChgd
390H	912	IA32_PERF_GLOBAL_STATUS_RESET		See Table 2-2. See Section 20.2.4, "Architectural Performance Monitoring Version 4."
		0	Thread	Set 1 to clear Ovf_PMCO.
		1	Thread	Set 1 to clear Ovf_PMC1.
		2	Thread	Set 1 to clear Ovf_PMC2.
		3	Thread	Set 1 to clear Ovf_PMC3.
		4	Thread	Set 1 to clear Ovf_PMC4 (if CPUID.OAH:EAX[15:8] > 4).
		5	Thread	Set 1 to clear Ovf_PMC5 (if CPUID.OAH:EAX[15:8] > 5).
		6	Thread	Set 1 to clear Ovf_PMC6 (if CPUID.OAH:EAX[15:8] > 6).
		7	Thread	Set 1 to clear Ovf_PMC7 (if CPUID.0AH:EAX[15:8] > 7).
		31:8		Reserved
		32	Thread	Set 1 to clear Ovf_FixedCtr0.
		33	Thread	Set 1 to clear Ovf_FixedCtr1.
		34	Thread	Set 1 to clear Ovf_FixedCtr2.
		54:35		Reserved
		55	Thread	Set 1 to clear Trace_ToPA_PMI.
		57:56		Reserved
		58	Thread	Set 1 to clear LBR_Frz.
		59	Thread	Set 1 to clear CTR_Frz.
		60	Thread	Set 1 to clear ASCI.
		61	Thread	Set 1 to clear Ovf_Uncore.
		62	Thread	Set 1 to clear Ovf_BufDSSAVE.
		63	Thread	Set 1 to clear CondChgd.

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
391H	913	IA32_PERF_GLOBAL_STATUS_SET		See Table 2-2. See Section 20.2.4, "Architectural Performance Monitoring Version 4."
		0	Thread	Set 1 to cause Ovf_PMC0 = 1.
		1	Thread	Set 1 to cause Ovf_PMC1 = 1.
		2	Thread	Set 1 to cause Ovf_PMC2 = 1.
		3	Thread	Set 1 to cause Ovf_PMC3 = 1.
		4	Thread	Set 1 to cause Ovf_PMC4=1 (if CPUID.OAH:EAX[15:8] > 4).
		5	Thread	Set 1 to cause Ovf_PMC5=1 (if CPUID.OAH:EAX[15:8] > 5).
		6	Thread	Set 1 to cause Ovf_PMC6=1 (if CPUID.OAH:EAX[15:8] > 6).
		7	Thread	Set 1 to cause Ovf_PMC7=1 (if CPUID.OAH:EAX[15:8] > 7).
		31:8		Reserved
		32	Thread	Set 1 to cause Ovf_FixedCtr0 = 1.
		33	Thread	Set 1 to cause Ovf_FixedCtr1 = 1.
		34	Thread	Set 1 to cause Ovf_FixedCtr2 = 1.
		54:35		Reserved
		55	Thread	Set 1 to cause Trace_ToPA_PMI = 1.
		57:56		Reserved
		58	Thread	Set 1 to cause LBR_Frz = 1.
		59	Thread	Set 1 to cause CTR_Frz = 1.
		60	Thread	Set 1 to cause ASCI = 1.
		61	Thread	Set 1 to cause Ovf_Uncore.
		62	Thread	Set 1 to cause Ovf_BufDSSAVE.
		63		Reserved
392H	913	IA32_PERF_GLOBAL_INUSE	Thread	See Table 2-2.
3F7H	1015	MSR_PEBS_FRONTEND	Thread	FrontEnd Precise Event Condition Select (R/W)
		2:0		Event Code Select
		3		Reserved
		4		Event Code Select High
		7:5		Reserved
		19:8		IDQ_Bubble_Length Specifier
		22:20		IDQ_Bubble_Width Specifier
		63:23		Reserved

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
500H	1280	IA32_SGX_SVN_STATUS	Thread	Status and SVN Threshold of SGX Support for ACM (R/O)
		0		Lock
				See Section 38.11.3, "Interactions with Authenticated Code Modules (ACMs)".
		15:1		Reserved
		23:16		SGX_SVN_SINIT
				See Section 38.11.3, "Interactions with Authenticated Code Modules (ACMs)".
		63:24		Reserved
560H	1376	IA32_RTIT_OUTPUT_BASE	Thread	Trace Output Base Register (R/W)
				See Table 2-2.
561H	1377	IA32_RTIT_OUTPUT_MASK_PTRS	Thread	Trace Output Mask Pointers Register (R/W)
				See Table 2-2.
570H	1392	IA32_RTIT_CTL	Thread	Trace Control Register (R/W)
		0		TraceEn
		1		CYCEn
		2		OS
		3		User
		6:4		Reserved, must be zero.
		7		CR3Filter
		8		ToPA
		9		Writing 0 will #GP if also setting TraceEn. MTCEn
		10		
		11		TSCEn DisRETC
		12		
		13		Reserved, must be zero. BranchEn
		17:14		MTCFreq
		18		Reserved, must be zero.
		22:19		CycThresh
		23		Reserved, must be zero.
		27:24		PSBFreq
		31:28		Reserved, must be zero.
		35:32		ADDRO_CFG
		39:36		ADDR1_CFG
		59.30		עסטען_כו ע

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		63:40		Reserved, must be zero.
571H	1393	IA32_RTIT_STATUS	Thread	Tracing Status Register (R/W)
		0		FilterEn, writes ignored.
		1		ContexEn, writes ignored.
		2		TriggerEn, writes ignored.
		3		Reserved
		4		Error (R/W)
		5		Stopped
		31:6		Reserved, must be zero.
		48:32		PacketByteCnt
		63:49		Reserved, must be zero.
572H	1394	IA32_RTIT_CR3_MATCH	Thread	Trace Filter CR3 Match Register (R/W)
		4:0		Reserved
		63:5		CR3[63:5] value to match
580H	1408	IA32_RTIT_ADDRO_A	Thread	Region 0 Start Address (R/W)
		63:0		See Table 2-2.
581H	1409	IA32_RTIT_ADDRO_B	Thread	Region 0 End Address (R/W)
		63:0		See Table 2-2.
582H	1410	IA32_RTIT_ADDR1_A	Thread	Region 1 Start Address (R/W)
		63:0		See Table 2-2.
583H	1411	IA32_RTIT_ADDR1_B	Thread	Region 1 End Address (R/W)
		63:0		See Table 2-2.
639H	1593	MSR_PPO_ENERGY_STATUS	Package	PPO Energy Status (R/O)
				See Section 15.10.4, "PPO/PP1 RAPL Domains."
64DH	1613	MSR_PLATFORM_ENERGY_COUNTER	Platform	Platform Energy Counter (R/O) This MSR is valid only if both platform vendor hardware implementation and BIOS enablement support it. This MSR will read 0 if not valid.
		31:0		Total energy consumed by all devices in the platform that receive power from integrated power delivery mechanism, included platform devices are processor cores, SOC, memory, add-on or peripheral devices that get powered directly from the platform power delivery means. The energy units are specified in the MSR_RAPL_POWER_UNIT.Enery_Status_Unit.
		63:32		Reserved
64EH	1614	MSR_PPERF	Thread	Productive Performance Count (R/O)

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

Regi Add	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		63:0		Hardware's view of workload scalability. See Section 15.4.5.1.
64FH	1615	MSR_CORE_PERF_LIMIT_REASONS	Package	Indicator of Frequency Clipping in Processor Cores (R/W)
				(Frequency refers to processor core frequency.)
		0		PROCHOT Status (R0)
				When set, frequency is reduced below the operating system request due to assertion of external PROCHOT.
		1		Thermal Status (R0)
				When set, frequency is reduced below the operating system request due to a thermal event.
		3:2		Reserved
		4		Residency State Regulation Status (R0)
				When set, frequency is reduced below the operating system request due to residency state regulation limit.
		5		Running Average Thermal Limit Status (R0)
				When set, frequency is reduced below the operating system request due to Running Average Thermal Limit (RATL).
		6		VR Therm Alert Status (R0)
				When set, frequency is reduced below the operating system request due to a thermal alert from a processor Voltage Regulator (VR).
		7		VR Therm Design Current Status (R0)
				When set, frequency is reduced below the operating system request due to VR thermal design current limit.
		8		Other Status (R0)
				When set, frequency is reduced below the operating system request due to electrical or other constraints.
		9		Reserved
		10		Package/Platform-Level Power Limiting PL1 Status (R0)
				When set, frequency is reduced below the operating system request due to package/platform-level power limiting PL1.
		11		Package/Platform-Level PL2 Power Limiting Status (R0)
				When set, frequency is reduced below the operating system request due to package/platform-level power limiting PL2/PL3.

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

Regi Addı		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		12		Max Turbo Limit Status (R0) When set, frequency is reduced below the operating system request due to multi-core turbo limits.
		13		Turbo Transition Attenuation Status (R0)
				When set, frequency is reduced below the operating system request due to Turbo transition attenuation. This prevents performance degradation due to frequent operating ratio changes.
		15:14		Reserved
		16		PROCHOT Log
				When set, indicates that the PROCHOT Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		17		Thermal Log When set, indicates that the Thermal Status bit has asserted since the log bit was last cleared. This log bit will sepain set until cleared by settures.
				This log bit will remain set until cleared by software writing 0.
		19:18		Reserved.
		20		Residency State Regulation Log
				When set, indicates that the Residency State Regulation Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		21		Running Average Thermal Limit Log
				When set, indicates that the RATL Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		22		VR Therm Alert Log
				When set, indicates that the VR Therm Alert Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		23		VR Thermal Design Current Log
				When set, indicates that the VR TDC Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		24		Other Log When set, indicates that the Other Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.
		25		Reserved
		26		Package/Platform-Level PL1 Power Limiting Log When set, indicates that the Package or Platform Level PL1 Power Limiting Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		27		Package/Platform-Level PL2 Power Limiting Log
				When set, indicates that the Package or Platform Level PL2/PL3 Power Limiting Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		28		Max Turbo Limit Log
				When set, indicates that the Max Turbo Limit Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		29		Turbo Transition Attenuation Log
				When set, indicates that the Turbo Transition Attenuation Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		63:30		Reserved
652H	1618	MSR_PKG_HDC_CONFIG	Package	HDC Configuration (R/W)
		2:0		PKG_Cx_Monitor
				Configures Package Cx state threshold for MSR_PKG_HDC_DEEP_RESIDENCY.
		63: 3		Reserved
653H	1619	MSR_CORE_HDC_RESIDENCY	Соге	Core HDC Idle Residency (R/O)
		63:0		Core_Cx_Duty_Cycle_Cnt
655H	1621	MSR_PKG_HDC_SHALLOW_RESIDENCY	Package	Accumulate the cycles the package was in C2 state and at least one logical processor was in forced idle (R/O)
		63:0		Pkg_C2_Duty_Cycle_Cnt

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
656H	1622	MSR_PKG_HDC_DEEP_RESIDENCY	Package	Package Cx HDC Idle Residency (R/O)
		63:0		Pkg_Cx_Duty_Cycle_Cnt
658H	1624	MSR_WEIGHTED_CORE_CO	Package	Core-count Weighted CO Residency (R/O)
		63:0		Increment at the same rate as the TSC. The increment each cycle is weighted by the number of processor cores in the package that reside in CO. If N cores are simultaneously in CO, then each cycle the counter increments by N.
659H	1625	MSR_ANY_CORE_CO	Package	Any Core CO Residency (R/O)
		63:0		Increment at the same rate as the TSC. The increment each cycle is one if any processor core in the package is in CO.
65AH	1626	MSR_ANY_GFXE_CO	Package	Any Graphics Engine CO Residency (R/O)
		63:0		Increment at the same rate as the TSC. The increment each cycle is one if any processor graphic device's compute engines are in CO.
65BH	1627	MSR_CORE_GFXE_OVERLAP_CO	Package	Core and Graphics Engine Overlapped CO Residency (R/O)
		63:0		Increment at the same rate as the TSC. The increment each cycle is one if at least one compute engine of the processor graphics is in CO and at least one processor core in the package is also in CO.
65CH	1628	MSR_PLATFORM_POWER_LIMIT	Platform	Platform Power Limit Control (R/W-L)
				Allows platform BIOS to limit power consumption of the platform devices to the specified values. The Long Duration power consumption is specified via Platform_Power_Limit_1 and Platform_Power_Limit_1_Time. The Short Duration power consumption limit is specified via the Platform_Power_Limit_2 with duration chosen by the processor. The processor implements an exponential-weighted
				algorithm in the placement of the time windows.
		14:0		Platform Power Limit #1
				Average Power limit value which the platform must not exceed over a time window as specified by Power_Limit_1_TIME field.
				The default value is the Thermal Design Power (TDP) and varies with product skus. The unit is specified in MSR_RAPLPOWER_UNIT.

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	7		
		15		Enable Platform Power Limit #1
				When set, enables the processor to apply control policy such that the platform power does not exceed Platform Power limit #1 over the time window specified by Power Limit #1 Time Window.
		16		Platform Clamping Limitation #1
				When set, allows the processor to go below the OS requested P states in order to maintain the power below specified Platform Power Limit #1 value.
				This bit is writeable only when CPUID (EAX=6):EAX[4] is set.
		23:17		Time Window for Platform Power Limit #1
				Specifies the duration of the time window over which Platform Power Limit 1 value should be maintained for sustained long duration. This field is made up of two numbers from the following equation:
				Time Window = (float) $((1+(X/4))*(2^Y))$, where:
				X = POWER_LIMIT_1_TIME[23:22]
				Y = POWER_LIMIT_1_TIME[21:17]
				The maximum allowed value in this field is defined in MSR_PKG_POWER_INFO[PKG_MAX_WIN].
				The default value is ODH, The unit is specified in MSR_RAPLPOWER_UNIT[Time Unit].
		31:24		Reserved
		46:32		Platform Power Limit #2
				Average Power limit value which the platform must not exceed over the Short Duration time window chosen by the processor.
				The recommended default value is 1.25 times the Long Duration Power Limit (i.e., Platform Power Limit # 1).
		47		Enable Platform Power Limit #2
				When set, enables the processor to apply control policy such that the platform power does not exceed Platform Power limit #2 over the Short Duration time window.
		48		Platform Clamping Limitation #2
				When set, allows the processor to go below the OS requested P states in order to maintain the power below specified Platform Power Limit #2 value.
		62:49		Reserved
		63		Lock. Setting this bit will lock all other bits of this MSR until system RESET.

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
690H	1680	MSR_LASTBRANCH_16_FROM_IP	Thread	Last Branch Record 16 From IP (R/W) One of 32 triplets of last branch record registers on the last branch record stack. This part of the stack contains pointers to the source instruction. See also: Last Branch Record Stack TOS at 1C9H. Section 18.12.
691H	1681	MSR_LASTBRANCH_17_FROM_IP	Thread	Last Branch Record 17 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
692H	1682	MSR_LASTBRANCH_18_FROM_IP	Thread	Last Branch Record 18 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
693H	1683	MSR_LASTBRANCH_19_FROM_IP	Thread	Last Branch Record 19From IP (R/W) See description of MSR_LASTBRANCH_O_FROM_IP.
694H	1684	MSR_LASTBRANCH_20_FROM_IP	Thread	Last Branch Record 20 From IP (R/W) See description of MSR_LASTBRANCH_O_FROM_IP.
695H	1685	MSR_LASTBRANCH_21_FROM_IP	Thread	Last Branch Record 21 From IP (R/W) See description of MSR_LASTBRANCH_O_FROM_IP.
696H	1686	MSR_LASTBRANCH_22_FROM_IP	Thread	Last Branch Record 22 From IP (R/W) See description of MSR_LASTBRANCH_O_FROM_IP.
697H	1687	MSR_LASTBRANCH_23_FROM_IP	Thread	Last Branch Record 23 From IP (R/W) See description of MSR_LASTBRANCH_O_FROM_IP.
698H	1688	MSR_LASTBRANCH_24_FROM_IP	Thread	Last Branch Record 24 From IP (R/W) See description of MSR_LASTBRANCH_O_FROM_IP.
699H	1689	MSR_LASTBRANCH_25_FROM_IP	Thread	Last Branch Record 25 From IP (R/W) See description of MSR_LASTBRANCH_O_FROM_IP.
69AH	1690	MSR_LASTBRANCH_26_FROM_IP	Thread	Last Branch Record 26 From IP (R/W) See description of MSR_LASTBRANCH_O_FROM_IP.
69BH	1691	MSR_LASTBRANCH_27_FROM_IP	Thread	Last Branch Record 27 From IP (R/W) See description of MSR_LASTBRANCH_O_FROM_IP.
69CH	1692	MSR_LASTBRANCH_28_FROM_IP	Thread	Last Branch Record 28 From IP (R/W) See description of MSR_LASTBRANCH_O_FROM_IP.
69DH	1693	MSR_LASTBRANCH_29_FROM_IP	Thread	Last Branch Record 29 From IP (R/W) See description of MSR_LASTBRANCH_O_FROM_IP.
69EH	1694	MSR_LASTBRANCH_30_FROM_IP	Thread	Last Branch Record 30 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.
69FH	1695	MSR_LASTBRANCH_31_FROM_IP	Thread	Last Branch Record 31 From IP (R/W) See description of MSR_LASTBRANCH_0_FROM_IP.

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

Regi Add	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
6B0H	1712	MSR_GRAPHICS_PERF_LIMIT_REASONS	Package	Indicator of Frequency Clipping in the Processor Graphics (R/W)
				(Frequency refers to processor graphics frequency.)
		0		PROCHOT Status (R0)
				When set, frequency is reduced due to assertion of external PROCHOT.
		1		Thermal Status (R0)
				When set, frequency is reduced due to a thermal event.
		4:2		Reserved.
		5		Running Average Thermal Limit Status (R0)
				When set, frequency is reduced due to running average thermal limit.
		6		VR Therm Alert Status (R0)
				When set, frequency is reduced due to a thermal alert from a processor Voltage Regulator.
		7		VR Thermal Design Current Status (R0)
				When set, frequency is reduced due to VR TDC limit.
		8		Other Status (R0)
				When set, frequency is reduced due to electrical or other constraints.
		9		Reserved
		10		Package/Platform-Level Power Limiting PL1 Status (R0)
				When set, frequency is reduced due to package/platform-level power limiting PL1.
		11		Package/Platform-Level PL2 Power Limiting Status (R0)
				When set, frequency is reduced due to package/platform-level power limiting PL2/PL3.
		12		Inefficient Operation Status (R0)
				When set, processor graphics frequency is operating below target frequency.
		15:13		Reserved
		16		PROCHOT Log
				When set, indicates that the PROCHOT Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

Regi Addı		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		17		Thermal Log When set, indicates that the Thermal Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software
				writing 0.
		20:18		Reserved.
		21		Running Average Thermal Limit Log
				When set, indicates that the RATL Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		22		VR Therm Alert Log
				When set, indicates that the VR Therm Alert Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		23		VR Thermal Design Current Log
				When set, indicates that the VR Therm Alert Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		24		Other Log
				When set, indicates that the OTHER Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		25		Reserved
		26		Package/Platform-Level PL1 Power Limiting Log
				When set, indicates that the Package/Platform Level PL1 Power Limiting Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		27		Package/Platform-Level PL2 Power Limiting Log
				When set, indicates that the Package/Platform Level PL2 Power Limiting Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec]		
		28		Inefficient Operation Log
				When set, indicates that the Inefficient Operation Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		63:29		Reserved
6B1H	1713	MSR_RING_PERF_LIMIT_REASONS	Package	Indicator of Frequency Clipping in the Ring Interconnect (R/W)
				(Frequency refers to ring interconnect in the uncore.)
		0		PROCHOT Status (R0)
				When set, frequency is reduced due to assertion of external PROCHOT.
		1		Thermal Status (R0)
				When set, frequency is reduced due to a thermal event.
		4:2		Reserved
		5		Running Average Thermal Limit Status (R0)
				When set, frequency is reduced due to running average thermal limit.
		6		VR Therm Alert Status (R0)
				When set, frequency is reduced due to a thermal alert from a processor Voltage Regulator.
		7		VR Thermal Design Current Status (R0)
				When set, frequency is reduced due to VR TDC limit.
		8		Other Status (R0)
				When set, frequency is reduced due to electrical or other constraints.
		9		Reserved
		10		Package/Platform-Level Power Limiting PL1 Status (R0)
				When set, frequency is reduced due to package/Platform-level power limiting PL1.
		11		Package/Platform-Level PL2 Power Limiting Status (R0)
				When set, frequency is reduced due to package/Platform-level power limiting PL2/PL3.
		15:12		Reserved

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

Regi: Addr		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		16		PROCHOT Log When set, indicates that the PROCHOT Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		17		Thermal Log
				When set, indicates that the Thermal Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		20:18		Reserved
		21		Running Average Thermal Limit Log
				When set, indicates that the RATL Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		22		VR Therm Alert Log
				When set, indicates that the VR Therm Alert Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		23		VR Thermal Design Current Log
				When set, indicates that the VR Therm Alert Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		24		Other Log
				When set, indicates that the OTHER Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.
		25		Reserved
		26		Package/Platform-Level PL1 Power Limiting Log
				When set, indicates that the Package/Platform Level PL1 Power Limiting Status bit has asserted since the log bit was last cleared.
				This log bit will remain set until cleared by software writing 0.

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		27		Package/Platform-Level PL2 Power Limiting Log When set, indicates that the Package/Platform Level PL2 Power Limiting Status bit has asserted since the log bit was last cleared. This log bit will remain set until cleared by software writing 0.
		63:28		Reserved
6D0H	1744	MSR_LASTBRANCH_16_TO_IP	Thread	Last Branch Record 16 To IP (R/W)
				One of 32 triplets of last branch record registers on the last branch record stack. This part of the stack contains pointers to the destination instruction. See also: Last Branch Record Stack TOS at 1C9H. Section 18.12.
6D1H	1745	MSR_LASTBRANCH_17_TO_IP	Thread	Last Branch Record 17 To IP (R/W)
				See description of MSR_LASTBRANCH_0_T0_IP.
6D2H	1746	MSR_LASTBRANCH_18_TO_IP	Thread	Last Branch Record 18 To IP (R/W)
				See description of MSR_LASTBRANCH_0_T0_IP.
6D3H	1747	MSR_LASTBRANCH_19_TO_IP	Thread	Last Branch Record 19To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6D4H	1748	MSR_LASTBRANCH_20_TO_IP	Thread	Last Branch Record 20 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6D5H	1749	MSR_LASTBRANCH_21_TO_IP	Thread	Last Branch Record 21 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6D6H	1750	MSR_LASTBRANCH_22_TO_IP	Thread	Last Branch Record 22 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6D7H	1751	MSR_LASTBRANCH_23_TO_IP	Thread	Last Branch Record 23 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6D8H	1752	MSR_LASTBRANCH_24_TO_IP	Thread	Last Branch Record 24 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6D9H	1753	MSR_LASTBRANCH_25_TO_IP	Thread	Last Branch Record 25 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6DAH	1754	MSR_LASTBRANCH_26_TO_IP	Thread	Last Branch Record 26 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6DBH	1755	MSR_LASTBRANCH_27_TO_IP	Thread	Last Branch Record 27 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6DCH	1756	MSR_LASTBRANCH_28_TO_IP	Thread	Last Branch Record 28 To IP (R/W) See description of MSR_LASTBRANCH_0_T0_IP.
6DDH	1757	MSR_LASTBRANCH_29_TO_IP	Thread	Last Branch Record 29 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

Regi Add	ster ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
6DEH	1758	MSR_LASTBRANCH_30_TO_IP	Thread	Last Branch Record 30 To IP (R/W) See description of MSR_LASTBRANCH_0_TO_IP.
6DFH	1759	MSR_LASTBRANCH_31_TO_IP	Thread	Last Branch Record 31 To IP (R/W)
				See description of MSR_LASTBRANCH_0_TO_IP.
770H	1904	IA32_PM_ENABLE	Package	See Section 15.4.2, "Enabling HWP".
771H	1905	IA32_HWP_CAPABILITIES	Thread	See Section 15.4.3, "HWP Performance Range and Dynamic Capabilities".
772H	1906	IA32_HWP_REQUEST_PKG	Package	See Section 15.4.4, "Managing HWP".
773H	1907	IA32_HWP_INTERRUPT	Thread	See Section 15.4.6, "HWP Notifications".
774H	1908	IA32_HWP_REQUEST	Thread	See Section 15.4.4, "Managing HWP".
		7:0		Minimum Performance (R/W)
		15:8		Maximum Performance (R/W)
		23:16		Desired Performance (R/W)
		31:24		Energy/Performance Preference (R/W)
		41:32		Activity Window (R/W)
		42		Package Control (R/W)
		63:43		Reserved
777H	1911	IA32_HWP_STATUS	Thread	See Section 15.4.5, "HWP Feedback".
D90H	3472	IA32_BNDCFGS	Thread	See Table 2-2.
DAOH	3488	IA32_XSS	Thread	See Table 2-2.
DBOH	3504	IA32_PKG_HDC_CTL	Package	See Section 15.5.2, "Package level Enabling HDC".
DB1H	3505	IA32_PM_CTL1	Thread	See Section 15.5.3, "Logical-Processor Level HDC Control".
DB2H	3506	IA32_THREAD_STALL	Thread	See Section 15.5.4.1, "IA32_THREAD_STALL".
DCOH	3520	MSR_LBR_INFO_0	Thread	Last Branch Record O Additional Information (R/W)
				One of 32 triplet of last branch record registers on the last branch record stack. This part of the stack contains flag, TSX-related and elapsed cycle information. See also:
				Last Branch Record Stack TOS at 1C9H.Section 18.9.1, "LBR Stack."
DC1H	3521	MSR_LBR_INFO_1	Thread	Last Branch Record 1 Additional Information (R/W)
				See description of MSR_LBR_INFO_0.
DC2H	3522	MSR_LBR_INFO_2	Thread	Last Branch Record 2 Additional Information (R/W)
				See description of MSR_LBR_INFO_0.
DC3H	3523	MSR_LBR_INFO_3	Thread	Last Branch Record 3 Additional Information (R/W)
				See description of MSR_LBR_INFO_0.

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
DC4H	3524	MSR_LBR_INFO_4	Thread	Last Branch Record 4 Additional Information (R/W) See description of MSR_LBR_INFO_0.
DC5H	3525	MSR_LBR_INFO_5	Thread	Last Branch Record 5 Additional Information (R/W) See description of MSR_LBR_INFO_0.
DC6H	3526	MSR_LBR_INFO_6	Thread	Last Branch Record 6 Additional Information (R/W) See description of MSR_LBR_INFO_0.
DC7H	3527	MSR_LBR_INFO_7	Thread	Last Branch Record 7 Additional Information (R/W) See description of MSR_LBR_INFO_0.
DC8H	3528	MSR_LBR_INFO_8	Thread	Last Branch Record 8 Additional Information (R/W) See description of MSR_LBR_INFO_0.
DC9H	3529	MSR_LBR_INFO_9	Thread	Last Branch Record 9 Additional Information (R/W) See description of MSR_LBR_INFO_0.
DCAH	3530	MSR_LBR_INFO_10	Thread	Last Branch Record 10 Additional Information (R/W) See description of MSR_LBR_INFO_0.
DCBH	3531	MSR_LBR_INFO_11	Thread	Last Branch Record 11 Additional Information (R/W) See description of MSR_LBR_INFO_0.
DCCH	3532	MSR_LBR_INFO_12	Thread	Last Branch Record 12 Additional Information (R/W) See description of MSR_LBR_INFO_0.
DCDH	3533	MSR_LBR_INFO_13	Thread	Last Branch Record 13 Additional Information (R/W) See description of MSR_LBR_INFO_0.
DCEH	3534	MSR_LBR_INFO_14	Thread	Last Branch Record 14 Additional Information (R/W) See description of MSR_LBR_INFO_0.
DCFH	3535	MSR_LBR_INFO_15	Thread	Last Branch Record 15 Additional Information (R/W) See description of MSR_LBR_INFO_0.
DD0H	3536	MSR_LBR_INFO_16	Thread	Last Branch Record 16 Additional Information (R/W) See description of MSR_LBR_INFO_0.
DD1H	3537	MSR_LBR_INFO_17	Thread	Last Branch Record 17 Additional Information (R/W) See description of MSR_LBR_INFO_0.
DD2H	3538	MSR_LBR_INFO_18	Thread	Last Branch Record 18 Additional Information (R/W) See description of MSR_LBR_INFO_0.
DD3H	3539	MSR_LBR_INFO_19	Thread	Last Branch Record 19 Additional Information (R/W) See description of MSR_LBR_INFO_0.
DD4H	3520	MSR_LBR_INFO_20	Thread	Last Branch Record 20 Additional Information (R/W) See description of MSR_LBR_INFO_0.
DD5H	3521	MSR_LBR_INFO_21	Thread	Last Branch Record 21 Additional Information (R/W) See description of MSR_LBR_INFO_0.

Table 2-39. Additional MSRs Supported by 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core™ Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processor Family, 8th Generation Intel® Core™ i3 Processors, and Intel® Xeon® E Processors

Regi Add	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
DD6H	3522	MSR_LBR_INFO_22	Thread	Last Branch Record 22 Additional Information (R/W)
				See description of MSR_LBR_INFO_0.
DD7H	3523	MSR_LBR_INFO_23	Thread	Last Branch Record 23 Additional Information (R/W)
				See description of MSR_LBR_INFO_0.
DD8H	3524	MSR_LBR_INFO_24	Thread	Last Branch Record 24 Additional Information (R/W)
				See description of MSR_LBR_INFO_0.
DD9H	3525	MSR_LBR_INFO_25	Thread	Last Branch Record 25 Additional Information (R/W)
				See description of MSR_LBR_INFO_0.
DDAH	3526	MSR_LBR_INFO_26	Thread	Last Branch Record 26 Additional Information (R/W)
				See description of MSR_LBR_INFO_0.
DDBH	3527	MSR_LBR_INFO_27	Thread	Last Branch Record 27 Additional Information (R/W)
				See description of MSR_LBR_INFO_0.
DDCH	3528	MSR_LBR_INFO_28	Thread	Last Branch Record 28 Additional Information (R/W)
				See description of MSR_LBR_INFO_0.
DDDH	3529	MSR_LBR_INFO_29	Thread	Last Branch Record 29 Additional Information (R/W)
				See description of MSR_LBR_INFO_0.
DDEH	3530	MSR_LBR_INFO_30	Thread	Last Branch Record 30 Additional Information (R/W)
				See description of MSR_LBR_INFO_0.
DDFH	3531	MSR_LBR_INFO_31	Thread	Last Branch Record 31 Additional Information (R/W)
				See description of MSR_LBR_INFO_0.

Table 2-40 lists the MSRs of uncore PMU for Intel processors with CPUID DisplayFamily_DisplayModel signatures of 06_4EH, 06_5EH, 06_8EH, 06_9EH, and 06_66H.

Table 2-40. Uncore PMU MSRs Supported by 6th Generation, 7th Generation, and 8th Generation Intel® Core™ Processors, and 8th generation Intel® Core™ i3 Processors

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
394H	916	MSR_UNC_PERF_FIXED_CTRL	Package	Uncore Fixed Counter Control (R/W)
		19:0		Reserved
		20		Enable overflow propagation.
		21		Reserved
		22		Enable counting.
		63:23		Reserved
395H	917	MSR_UNC_PERF_FIXED_CTR	Package	Uncore Fixed Counter
		43:0		Current count.

Table 2-40. Uncore PMU MSRs Supported by 6th Generation, 7th Generation, and 8th Generation Intel® Core™ Processors, and 8th generation Intel® Core™ i3 Processors

Regi Addı		Register Name / Bit Fields	Scope	© Core in i3 Processors Bit Description
Hex	Dec			
		63:44		Reserved
396H	918	MSR_UNC_CBO_CONFIG	Package	Uncore C-Box Configuration Information (R/O)
		3:0		Specifies the number of C-Box units with programmable counters (including processor cores and processor graphics).
		63:4		Reserved
3B0H	946	MSR_UNC_ARB_PERFCTR0	Package	Uncore Arb Unit, Performance Counter 0
3B1H	947	MSR_UNC_ARB_PERFCTR1	Package	Uncore Arb Unit, Performance Counter 1
3B2H	944	MSR_UNC_ARB_PERFEVTSEL0	Package	Uncore Arb Unit, Counter 0 Event Select MSR
3B3H	945	MSR_UNC_ARB_PERFEVTSEL1	Package	Uncore Arb Unit, Counter 1 Event Select MSR
700H	1792	MSR_UNC_CBO_O_PERFEVTSELO	Package	Uncore C-Box O, Counter O Event Select MSR
701H	1793	MSR_UNC_CBO_O_PERFEVTSEL1	Package	Uncore C-Box O, Counter 1 Event Select MSR
706H	1798	MSR_UNC_CBO_O_PERFCTRO	Package	Uncore C-Box O, Performance Counter O
707H	1799	MSR_UNC_CBO_0_PERFCTR1	Package	Uncore C-Box O, Performance Counter 1
710H	1808	MSR_UNC_CBO_1_PERFEVTSELO	Package	Uncore C-Box 1, Counter 0 Event Select MSR
711H	1809	MSR_UNC_CBO_1_PERFEVTSEL1	Package	Uncore C-Box 1, Counter 1 Event Select MSR
716H	1814	MSR_UNC_CBO_1_PERFCTRO	Package	Uncore C-Box 1, Performance Counter 0
717H	1815	MSR_UNC_CBO_1_PERFCTR1	Package	Uncore C-Box 1, Performance Counter 1
720H	1824	MSR_UNC_CBO_2_PERFEVTSELO	Package	Uncore C-Box 2, Counter 0 Event Select MSR
721H	1825	MSR_UNC_CBO_2_PERFEVTSEL1	Package	Uncore C-Box 2, Counter 1 Event Select MSR
726H	1830	MSR_UNC_CBO_2_PERFCTRO	Package	Uncore C-Box 2, Performance Counter 0
727H	1831	MSR_UNC_CBO_2_PERFCTR1	Package	Uncore C-Box 2, Performance Counter 1
730H	1840	MSR_UNC_CBO_3_PERFEVTSELO	Package	Uncore C-Box 3, Counter 0 Event Select MSR
731H	1841	MSR_UNC_CBO_3_PERFEVTSEL1	Package	Uncore C-Box 3, Counter 1 Event Select MSR
736H	1846	MSR_UNC_CBO_3_PERFCTRO	Package	Uncore C-Box 3, Performance Counter 0
737H	1847	MSR_UNC_CBO_3_PERFCTR1	Package	Uncore C-Box 3, Performance Counter 1
E01H	3585	MSR_UNC_PERF_GLOBAL_CTRL	Package	Uncore PMU Global Control
		0		Slice 0 select.
		1		Slice 1 select.
		2		Slice 2 select.
		3		Slice 3 select.
		4		Slice 4select.
		18:5		Reserved
		29		Enable all uncore counters.
		30		Enable wake on PMI.
		31		Enable Freezing counter when overflow.
		63:32		Reserved

Table 2-40. Uncore PMU MSRs Supported by 6th Generation, 7th Generation, and 8th Generation Intel® Core™ Processors, and 8th generation Intel® Core™ i3 Processors

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
E02H	3586	MSR_UNC_PERF_GLOBAL_STATUS	Package	Uncore PMU Main Status
		0		Fixed counter overflowed.
		1		An ARB counter overflowed.
		2		Reserved
		3		A CBox counter overflowed (on any slice).
		63:4		Reserved

2.17.1 MSRs Introduced in 7th Generation and 8th Generation Intel® Core™ Processors based on Kaby Lake Microarchitecture and Coffee Lake Microarchitecture

Table 2-41 lists additional MSRs for 7th generation and 8th generation Intel Core processors with a CPUID DisplayFamily_DisplayModel signatures of 06_8EH and 06_9EH. For an MSR listed in Table 2-41 that also appears in the model-specific tables of prior generations, Table 2-41 supersedes prior generation tables.

Table 2-41. Additional MSRs Supported by 7th Generation and 8th Generation Intel® Core™ Processors

Based on Kaby Lake Microarchitecture and Coffee Lake Microarchitecture

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
80H	128	MSR_TRACE_HUB_STH_ACPIBAR_BASE	Package	NPK Address Used by AET Messages (R/W)
		0		Lock Bit
				If set, this MSR cannot be re-written anymore. Lock bit has to be set in order for the AET packets to be directed to NPK MMIO.
		17:1		Reserved
		63:18		ACPIBAR_BASE_ADDRESS
				AET target address in NPK MMIO space.
1F4H	500	MSR_PRMRR_PHYS_BASE	Core	Processor Reserved Memory Range Register - Physical Base Control Register (R/W)
		2:0		MemType
				PRMRR BASE MemType.
		11:3		Reserved
		45:12		Base
				PRMRR Base Address.
		63:46		Reserved
1F5H	501	MSR_PRMRR_PHYS_MASK	Соге	Processor Reserved Memory Range Register - Physical Mask Control Register (R/W)
		9:0		Reserved
		10		Lock
				Lock bit for the PRMRR.

Table 2-41. Additional MSRs Supported by 7th Generation and 8th Generation Intel® Core™ Processors Based on Kaby Lake Microarchitecture and Coffee Lake Microarchitecture

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	1		
		11		VLD Enable bit for the PRMRR.
		45:12		Mask PRMRR MASK bits.
		63:46		Reserved
1FBH	507	MSR_PRMRR_VALID_CONFIG	Соге	Valid PRMRR Configurations (R/W)
		0		1M supported MEE size.
		4:1		Reserved
		5		32M supported MEE size.
		6		64M supported MEE size.
		7		128M supported MEE size.
		31:8		Reserved
2F4H	756	MSR_UNCORE_PRMRR_PHYS_BASE ¹	Package	(R/W) The PRMRR range is used to protect the processor reserved memory from unauthorized reads and writes. Any IO access to this range is aborted. This register controls the location of the PRMRR range by indicating its starting address. It functions in tandem with the PRMRR mask register.
		11:0		Reserved
		PAWIDTH-1:12		Range Base This field corresponds to bits PAWIDTH-1:12 of the base address memory range which is allocated to PRMRR memory.
		63:PAWIDTH		Reserved
2F5H	757	MSR_UNCORE_PRMRR_PHYS_MASK ¹	Package	(R/W) This register controls the size of the PRMRR range by indicating which address bits must match the PRMRR base register value.
		9:0		Reserved
		10		Lock
				Setting this bit locks all writeable settings in this register, including itself.
		11		Range_En Indicates whether the PRMRR range is enabled and valid.
		38:12		Range_Mask This field indicates which address bits must match PRMRR base in order to qualify as an PRMRR access.
		63:39		Reserved

Table 2-41. Additional MSRs Supported by 7th Generation and 8th Generation Intel® Core™ Processors

Based on Kaby Lake Microarchitecture and Coffee Lake Microarchitecture

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
620H	1568	MSR_RING_RATIO_LIMIT	Package	Ring Ratio Limit (R/W)
				This register provides Min/Max Ratio Limits for the LLC and Ring.
		6:0		MAX_Ratio
				This field is used to limit the max ratio of the LLC/Ring.
		7		Reserved
		14:8		MIN_Ratio
				Writing to this field controls the minimum possible ratio of the LLC/Ring.
		63:15		Reserved

NOTES:

2.17.2 MSRs Specific to 8th Generation Intel® Core™ i3 Processors

Table 2-42 lists additional MSRs for 8th generation Intel Core i3 processors with a CPUID DisplayFamily_DisplayModel signature of 06_66H. For an MSR listed in Table 2-42 that also appears in the model-specific tables of prior generations, Table 2-42 supersede prior generation tables.

Table 2-42. Additional MSRs Supported by 8th Generation Intel® Core™ i3 Processors

Based on Cannon Lake Microarchitecture

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
ЗАН	58	IA32_FEATURE_CONTROL	Thread	Control Features in Intel 64 Processor (R/W)
				See Table 2-2.
		0		Lock (R/WL)
		1		Enable VMX Inside SMX Operation (R/WL)
		2		Enable VMX Outside SMX Operation (R/WL)
		14:8		SENTER Local Functions Enables (R/WL)
		15		SENTER Global Functions Enable (R/WL)
		17		SGX Launch Control Enable (R/WL)
				This bit must be set to enable runtime reconfiguration of SGX Launch Control via IA32_SGXLEPUBKEYHASHN MSR.
				Available only if CPUID.(EAX=07H, ECX=0H): ECX[30] = 1.
		18		SGX Global Functions Enable (R/WL)
		63:21		Reserved
350H	848	MSR_BR_DETECT_CTRL		Branch Monitoring Global Control (R/W)

^{1.} This MSR is specific to 7th generation and 8th generation Intel® Core™ processors.

Table 2-42. Additional MSRs Supported by 8th Generation Intel® Core™ i3 Processors Based on Cannon Lake Microarchitecture (Contd.)

Regi Add	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		0		EnMonitoring
				Global enable for branch monitoring.
		1		EnExcept
				Enable branch monitoring event signaling on threshold trip.
				The branch monitoring event handler is signaled via the existing PMI signaling mechanism as programmed from the corresponding local APIC LVT entry.
		2		EnLBRFrz
				Enable LBR freeze on threshold trip. This will cause the LBR frozen bit 58 to be set in IA32_PERF_GLOBAL_STATUS when a triggering condition occurs and this bit is enabled.
		3		DisableInGuest
				When set to '1', branch monitoring, event triggering and LBR freeze actions are disabled when operating at VMX non-root operation.
		7:4		Reserved
		17:8		WindowSize
				Window size defined by WindowCntSel. Values 0 – 1023 are supported.
				Once the Window counter reaches the WindowSize count both the Window Counter and all Branch Monitoring Counters are cleared.
		23:18		Reserved
		25:24		WindowCntSel
				Window event count select:
				'00 = Instructions retired.
				'01 = Branch instructions retired
				'10 = Return instructions retired.
				'11 = Indirect branch instructions retired.
		26		CntAndMode
				When set to '1', the overall branch monitoring event triggering condition is true only if all enabled counters' threshold conditions are true.
				When '0', the threshold tripping condition is true if any enabled counters' threshold is true.
		63:27		Reserved
351H	849	MSR_BR_DETECT_STATUS		Branch Monitoring Global Status (R/W)
		0		Branch Monitoring Event Signaled
				When set to '1', Branch Monitoring event signaling is blocked until this bit is cleared by software.

Table 2-42. Additional MSRs Supported by 8th Generation Intel® Core™ i3 Processors Based on Cannon Lake Microarchitecture (Contd.)

Regi Addı		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		1		LBRsValid This status bit is set to '1' if the LBR state is considered valid for sampling by branch monitoring software.
		7:2		Reserved
		8		CntrHit0
				Branch monitoring counter #0 threshold hit. This status bit is sticky and once set requires clearing by software. Counter operation continues independent of the state of the bit.
		9		CntrHit1
				Branch monitoring counter #1 threshold hit. This status bit is sticky and once set requires clearing by software. Counter operation continues independent of the state of the bit.
		15:10		Reserved
				Reserved for additional branch monitoring counters threshold hit status.
		25:16		CountWindow
				The current value of the window counter. The count value is frozen on a valid branch monitoring triggering condition. This is a 10-bit unsigned value.
		31:26		Reserved
				Reserved for future extension of CountWindow.
		39:32		Count0
				The current value of counter 0 updated after each occurrence of the event being counted. The count value is frozen on a valid branch monitoring triggering condition (in which case CntrHitO will also be set). This is an 8-bit signed value (2's complement).
				Heuristic events which only increment will saturate and freeze at maximum value 0xFF (256).
				RET-CALL event counter saturate at maximum value 0x7F (+127) and minimum value 0x80 (-128).
		47:40		Count1
				The current value of counter 1 updated after each occurrence of the event being counted. The count value is frozen on a valid branch monitoring triggering condition (in which case CntrHit1 will also be set). This is an 8-bit signed value (2's complement).
				Heuristic events which only increment will saturate and freeze at maximum value 0xFF (256).
				RET-CALL event counter saturate at maximum value 0x7F (+127) and minimum value 0x80 (-128).
		63:48		Reserved

Table 2-42. Additional MSRs Supported by 8th Generation Intel® Core™ i3 Processors Based on Cannon Lake Microarchitecture (Contd.)

Regi Addı		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
354H - 355H	852 - 853	MSR_BR_DETECT_COUNTER_CONFIG_i		Branch Monitoring Detect Counter Configuration (R/W)
		0		CntrEn Enable counter.
		7:1		CntrEvSel Event select (other values #GP) '0000000 = RETs. '0000001 = RET-CALL bias. '0000010 = RET mispredicts. '0000011 = Branch (all) mispredicts. '0000100 = Indirect branch mispredicts. '0000101 = Far branch instructions.
		14:8		CntrThreshold Threshold (an unsigned value of 0 to 127 supported). The value 0 of counter threshold will result in event signaled after every instruction. #GP if threshold is < 2.
		15		MispredEventCnt Mispredict events counting behavior: '0 = Mispredict events are counted in a window. '1 = Mispredict events are counted based on a consecutive occurrence. CntrThreshold is treated as # of consecutive mispredicts. This control bit only applies to events specified by CntrEvSel that involve a prediction (0000010, 0000011, 0000100). Setting this bit for other events is ignored.
		63:16		Reserved
3F8H	1016	MSR_PKG_C3_RESIDENCY 63:0	Package	Package C3 Residency Counter (R/O) Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-states.
620H	1568	MSR_RING_RATIO_LIMIT	Package	Ring Ratio Limit (R/W) This register provides Min/Max Ratio Limits for the LLC and Ring.
		6:0		MAX_Ratio This field is used to limit the max ratio of the LLC/Ring.
		7		Reserved
		14:8		MIN_Ratio Writing to this field controls the minimum possible ratio of the LLC/Ring.
		63:15		Reserved

Table 2-42. Additional MSRs Supported by 8th Generation Intel® Core™ i3 Processors Based on Cannon Lake Microarchitecture (Contd.)

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
660H	1632	MSR_CORE_C1_RESIDENCY	Core	Core C1 Residency Counter (R/O)
		63:0		Value since last reset for the Core C1 residency. Counter rate is the Max Non-Turbo frequency (same as TSC). This counter counts in case both of the core's threads are in an idle state and at least one of the core's thread residency is in a C1 state or in one of its sub states. The counter is updated only after a core C state exit. Note: Always reads 0 if core C1 is unsupported. A value of zero indicates that this processor does not support core C1 or never entered core C1 level state.
662H	1634	MSR_CORE_C3_RESIDENCY	Core	Core C3 Residency Counter (R/O)
		63:0		Will always return 0.

Table 2-43 lists the MSRs of uncore PMU for Intel processors with CPUID signature 06_66H.

Table 2-43. Uncore PMU MSRs Supported by Intel® Core™ Processors Based on Cannon Lake Microarchitecture

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
394H	916	MSR_UNC_PERF_FIXED_CTRL	Package	Uncore Fixed Counter Control (R/W)
		19:0		Reserved
		20		Enable overflow propagation.
		21		Reserved
		22		Enable counting.
		63:23		Reserved
395H	917	MSR_UNC_PERF_FIXED_CTR	Package	Uncore Fixed Counter
		47:0		Current count.
		63:48		Reserved
396H	918	MSR_UNC_CBO_CONFIG	Package	Uncore C-Box Configuration Information (R/O)
		3:0		Report the number of C-Box units with performance counters, including processor cores and processor graphics.
		63:4		Reserved
3B0H	946	MSR_UNC_ARB_PERFCTR0	Package	Uncore Arb Unit, Performance Counter 0
3B1H	947	MSR_UNC_ARB_PERFCTR1	Package	Uncore Arb Unit, Performance Counter 1
3B2H	944	MSR_UNC_ARB_PERFEVTSEL0	Package	Uncore Arb Unit, Counter 0 Event Select MSR
3B3H	945	MSR_UNC_ARB_PERFEVTSEL1	Package	Uncore Arb unit, Counter 1 Event Select MSR
700H	1792	MSR_UNC_CBO_O_PERFEVTSELO	Package	Uncore C-Box 0, Counter 0 Event Select MSR

Table 2-43. Uncore PMU MSRs Supported by Intel® Core™ Processors Based on Cannon Lake Microarchitecture

Register Address		Register Name / Bit Fields	Scope Scope	Bit Description
Hex	Dec			
701H	1793	MSR_UNC_CBO_0_PERFEVTSEL1	Package	Uncore C-Box O, Counter 1 Event Select MSR
702H	1794	MSR_UNC_CBO_O_PERFCTRO	Package	Uncore C-Box 0, Performance Counter 0
703H	1795	MSR_UNC_CBO_0_PERFCTR1	Package	Uncore C-Box 0, Performance Counter 1
708H	1800	MSR_UNC_CBO_1_PERFEVTSELO	Package	Uncore C-Box 1, Counter 0 Event Select MSR
709H	1801	MSR_UNC_CBO_1_PERFEVTSEL1	Package	Uncore C-Box 1, Counter 1 Event Select MSR
70AH	1802	MSR_UNC_CBO_1_PERFCTRO	Package	Uncore C-Box 1, Performance Counter 0
70BH	1803	MSR_UNC_CBO_1_PERFCTR1	Package	Uncore C-Box 1, Performance Counter 1
710H	1808	MSR_UNC_CBO_2_PERFEVTSELO	Package	Uncore C-Box 2, Counter 0 Event Select MSR
711H	1809	MSR_UNC_CBO_2_PERFEVTSEL1	Package	Uncore C-Box 2, Counter 1 Event Select MSR
712H	1810	MSR_UNC_CBO_2_PERFCTRO	Package	Uncore C-Box 2, Performance Counter 0
713H	1811	MSR_UNC_CBO_2_PERFCTR1	Package	Uncore C-Box 2, Performance Counter 1
718H	1816	MSR_UNC_CBO_3_PERFEVTSEL0	Package	Uncore C-Box 3, Counter 0 Event Select MSR
719H	1817	MSR_UNC_CBO_3_PERFEVTSEL1	Package	Uncore C-Box 3, Counter 1 Event Select MSR
71AH	1818	MSR_UNC_CBO_3_PERFCTRO	Package	Uncore C-Box 3, Performance Counter 0
71BH	1819	MSR_UNC_CBO_3_PERFCTR1	Package	Uncore C-Box 3, Performance Counter 1
720H	1824	MSR_UNC_CBO_4_PERFEVTSEL0	Package	Uncore C-Box 4, Counter 0 Event Select MSR
721H	1825	MSR_UNC_CBO_4_PERFEVTSEL1	Package	Uncore C-Box 4, Counter 1 Event Select MSR
722H	1826	MSR_UNC_CBO_4_PERFCTRO	Package	Uncore C-Box 4, Performance Counter 0
723H	1827	MSR_UNC_CBO_4_PERFCTR1	Package	Uncore C-Box 4, Performance Counter 1
728H	1832	MSR_UNC_CBO_5_PERFEVTSEL0	Package	Uncore C-Box 5, Counter 0 Event Select MSR
729H	1833	MSR_UNC_CBO_5_PERFEVTSEL1	Package	Uncore C-Box 5, Counter 1 Event Select MSR
72AH	1834	MSR_UNC_CBO_5_PERFCTRO	Package	Uncore C-Box 5, Performance Counter 0
72BH	1835	MSR_UNC_CBO_5_PERFCTR1	Package	Uncore C-Box 5, Performance Counter 1
730H	1840	MSR_UNC_CBO_6_PERFEVTSEL0	Package	Uncore C-Box 6, Counter 0 Event Select MSR
731H	1841	MSR_UNC_CBO_6_PERFEVTSEL1	Package	Uncore C-Box 6, Counter 1 Event Select MSR
732H	1842	MSR_UNC_CBO_6_PERFCTRO	Package	Uncore C-Box 6, Performance Counter 0
733H	1843	MSR_UNC_CBO_6_PERFCTR1	Package	Uncore C-Box 6, Performance Counter 1
738H	1848	MSR_UNC_CBO_7_PERFEVTSEL0	Package	Uncore C-Box 7, Counter 0 Event Select MSR
739H	1849	MSR_UNC_CBO_7_PERFEVTSEL1	Package	Uncore C-Box 7, Counter 1 Event Select MSR
73AH	1850	MSR_UNC_CBO_7_PERFCTRO	Package	Uncore C-Box 7, Performance Counter 0
73BH	1851	MSR_UNC_CBO_7_PERFCTR1	Package	Uncore C-Box 7, Performance Counter 1
E01H	3585	MSR_UNC_PERF_GLOBAL_CTRL	Package	Uncore PMU Global Control
		0		Slice 0 select.
		1		Slice 1 select.
		2		Slice 2 select.
		3		Slice 3 select.
		4		Slice 4select.

Table 2-43. Uncore PMU MSRs Supported by Intel® Core™ Processors Based on Cannon Lake Microarchitecture

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		18:5		Reserved
		29		Enable all uncore counters.
		30		Enable wake on PMI.
		31		Enable Freezing counter when overflow.
		63:32		Reserved
E02H	3586	MSR_UNC_PERF_GLOBAL_STATUS	Package	Uncore PMU Main Status
		0		Fixed counter overflowed.
		1		An ARB counter overflowed.
		2		Reserved
		3		A CBox counter overflowed (on any slice).
		63:4		Reserved

2.17.3 MSRs Introduced in 10th Generation Intel® Core™ Processors

Table 2-44 lists additional MSRs for 10th generation Intel Core processors with a CPUID DisplayFamily_DisplayModel signature values of 06_7DH and 06_7EH. For an MSR listed in Table 2-44 that also appears in the model-specific tables of prior generations, Table 2-44 supersede prior generation tables.

Table 2-44. MSRs Supported by 10th Generation Intel® Core™ Processors Based on Ice Lake Microarchitecture

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
33H	51	MSR_MEMORY_CTRL	Соге	Memory Control Register
		28:0		Reserved.
		29		Enable #AC(0) exception for split locked accesses:
				Cause #AC(0) exception for split locked access at all CPL irrespective of CRO.AM or EFLAGS.AC.
		30		Reserved.
		31		Reserved.
48H	72	IA32_SPEC_CTRL	Соге	See Table 2-2.
49H	73	IA32_PREDICT_CMD	Thread	See Table 2-2.
8CH	140	IA32_SGXLEPUBKEYHASHO	Thread	See Table 2-2.
8DH	141	IA32_SGXLEPUBKEYHASH1	Thread	See Table 2-2.
8EH	142	IA32_SGXLEPUBKEYHASH2	Thread	See Table 2-2.
8FH	143	IA32_SGXLEPUBKEYHASH3	Thread	See Table 2-2.
AOH	160	MSR_BIOS_MCU_ERRORCODE	Package	BIOS MCU ERRORCODE (R/O)
				This MSR indicates if WRMSR 0x79 failed to configure PRM memory and gives a hint to debug BIOS.
		15:0	Package	Error Codes (R/O)

Table 2-44. MSRs Supported by 10th Generation Intel® Core™ Processors Based on Ice Lake Microarchitecture

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		30:16		Reserved.
		31	Thread	MCU Partial Success (R/O)
				When set to 1, WRMSR 0x79 skipped part of the functionality during BIOS.
A5H	165	MSR_FIT_BIOS_ERROR	Thread	FIT BIOS ERROR (R/W)
				Report error codes for debug in case the processor failed to parse the Firmware Table in BIOS.
				Can also be used to log BIOS information.
		7:0		Error Codes (R/W)
				Error codes for debug.
		15:8		Entry Type (R/W)
				Failed FIT entry type.
		16		FIT MCU Entry (R/W)
				FIT contains MCU entry.
		62:17		Reserved.
		63		LOCK (R/W)
				When set to 1, writes to this MSR will be skipped.
10BH	267	IA32_FLUSH_CMD	Thread	See Table 2-2.
151H	337	MSR_BIOS_DONE	Thread	BIOS Done (R/W0)
		0	Thread	BIOS Done Indication (R/WO)
				Set by BIOS when it finishes programming the processor and wants to lock the memory configuration from changes by software that is running on this thread.
				Writes to the bit will be ignored if EAX[0] is 0.
		1	Package	Package BIOS Done Indication (R/O)
		·	i dekage	When set to 1, all threads in the package have bit 0 of this MSR set.
		31:2		Reserved.
1F1H	497	MSR_CRASHLOG_CONTROL	Thread	Write Data to a Crash Log Configuration
		0		CDDIS: CrashDump_Disable
				If set, indicates that Crash Dump is disabled.
		63:1		Reserved.
2A0H	672	MSR_PRMRR_BASE_0	Core	Processor Reserved Memory Range Register - Physical Base Control Register (R/W)
		2:0		MEMTYPE: PRMRR BASE Memory Type.
		3		CONFIGURED: PRMRR BASE Configured.
		11:4		Reserved.
		51:12		BASE: PRMRR Base Address.
		63:52		Reserved.

Table 2-44. MSRs Supported by 10th Generation Intel® Core™ Processors Based on Ice Lake Microarchitecture

Hex Dec 30CH 780 IA32_FIXED_CTR3 Thread Fixed-Function Performance	
30CH 780 IA32_FIXED_CTR3 Thread Fixed-Function Perform	
Bit definitions are the IA32_FIXED_CTR0, of	mance Counter Register 3 (R/W) same as found in fset 309H. See Table 2-2.
329H 809 MSR_PERF_METRICS Thread Performance Metrics ((R/W)
expose to its guests) to PERF_METRICS feature	
7:0 Retiring. Percent of ut eventually retire (com	tilized slots by uops that mit).
incorrect speculation, o	ent of wasted slots due to covering utilized by uops that do v bubbles (unutilized slots).
	ent of unutilized slots where ver a uop while back-end is
	ent of unutilized slots where a I to back-end due to lack of back-
63:25 Reserved.	
3F2H 1010 MSR_PEBS_DATA_CFG Thread PEBS Data Configurati	ion (R/W)
of interest and thus re and record generation size and layout vary b	e capability to select data groups educe the record size in memory a latency. Hence, a PEBS record's based on the selected groups. coftware to select LBR depth for
O Memory Info.	
	pture memory information such data source and latency of the PEBS record.
1 GPRs.	
	pture the contents of the sters in the PEBS record.
Z XMMs.	
Setting this bit will cap registers in the PEBS in	pture the contents of the XMM record.
3 LBRs.	
Setting this bit will cap the PEBS record.	pture LBR TO, FROM, and INFO in

Table 2-44. MSRs Supported by 10th Generation Intel® Core™ Processors Based on Ice Lake Microarchitecture

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		31:24		LBR Entries. Set the field to the desired number of entries - 1. For example, if the LBR_entries field is 0, a single entry will be included in the record. To include 32 LBR entries, set the LBR_entries field to 31 (0x1F). To ensure all PEBS records are 16-byte aligned, software can use LBR_entries that is multiple of 3.
541H	1345	MSR_CORE_UARCH_CTL	Соге	Core Microarchitecture Control MSR (R/W)
		0		L1 Scrubbing Enable When set to 1, enable L1 scrubbing. Reserved.
CEZLI	1622	31:1	Thered	11222112
657H	1623	MSR_FAST_UNCORE_MSRS_CTL	Thread	Fast WRMSR/RDMSR Control MSR (R/W)
		3:0		FAST_ACCESS_ENABLE: Bit 0: When set to '1', provides a hint for the hardware to enable fast access mode for the IA32_HWP_REQUEST MSR.
				This bit is sticky and is cleaned by the hardware only during reset time.
				This bit is valid only if FAST_UNCORE_MSRS_CAPABILITY[0] is set. Setting this bit will cause CPUID[6].EAX[18] to be set.
		31:4		Reserved.
65EH	1630	MSR_FAST_UNCORE_MSRS_STATUS	Thread	Indication of Uncore MSRs, Post Write Activates
		0		Indicates whether the CPU is still in the middle of writing IA32_HWP_REQUEST MSR, even after the WRMSR instruction has retired.
				A value of 1 indicates the last write of IA32_HWP_REQUEST is still ongoing.
				A value of 0 indicates the last write of IA32_HWP_REQUEST is visible outside the logical processor.
				Software can use the status of this bit to avoid overwriting IA32_HWP_REQUEST.
		31:1		Reserved.
65FH	1631	MSR_FAST_UNCORE_MSRS_CAPABILITY	Thread	Fast WRMSR/RDMSR Enumeration MSR (R/O)
		3:0		MSRS_CAPABILITY:
				Bit 0: If set to '1', hardware supports the fast access mode for the IA32_HWP_REQUEST MSR.
		31:4		Reserved.
772H	1906	IA32_HWP_REQUEST_PKG	Package	See Table 2-2.
775H	1909	IA32_PECI_HWP_REQUEST_INFO	Thread	See Table 2-2.
777H	1911	IA32_HWP_STATUS	Thread	See Table 2-2.

2.17.4 MSRs Introduced in 11th Generation Intel® Core™ Processors based on Tiger Lake Microarchitecture

Table 2-45 lists additional MSRs for 11th generation Intel Core processors with CPUID DisplayFamily_DisplayModel signatures of 06_8CH and 06_8DH. The MSRs listed in Table 2-44 are also supported by these processors. For an MSR listed in Table 2-45 that also appears in the model-specific tables of prior generations, Table 2-45 supersedes prior generation tables.

Table 2-45. Additional MSRs Supported by 11th Generation Intel® Core™ Processors Based on Tiger Lake Microarchitecture

Regi: Addr		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
AOH	160	MSR_BIOS_MCU_ERRORCODE	Package	BIOS MCU ERRORCODE (R/O)
		15:0		Error Codes
		31:16		Reserved
A7H	167	MSR_BIOS_DEBUG	Thread	BIOS DEBUG (R/O)
				This MSR indicates if WRMSR 79H failed to configure PRM memory and gives a hint to debug BIOS.
		30:0		Reserved
		31		MCU Partial Success
				When set to 1, WRMSR 79H skipped part of the functionality during BIOS.
		63:32		Reserved
CFH	207	IA32_CORE_CAPABILITIES	Package	IA32 Core Capabilities Register (R/O)
				If CPUID.(EAX=07H, ECX=0):EDX[30] = 1.
				This MSR provides an architectural enumeration function for model-specific behavior.
		1:0		Reserved
		2		FUSA_SUPPORTED
		3		RSM_IN_CPLO_ONLY
				When set to 1, the RSM instruction is only allowed in CPLO (#GP triggered in any CPL != 0).
				When set to 0, then any CPL may execute the RSM instruction.
		4		Reserved
		5		SPLIT_LOCK_DISABLE_SUPPORTED
				When set to 1, the ability to set MEMORY_CONTROL (MSR 33H) bit 29 enables an #AC to be created when a split lock is detected.
		31:6		Reserved

Table 2-45. Additional MSRs Supported by 11th Generation Intel® Core™ Processors Based on Tiger Lake Microarchitecture

Regi Add	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
492H	1170	IA32_VMX_PROCBASED_CTLS3	Core	IA32_VMX_PROCBASED_CTLS3 This MSR enumerates the allowed 1-settings of the third set of processor-based controls. Specifically, VM entry allows bit X of the tertiary processor-based VM-execution controls to be 1 if and only if bit X of the MSR is set to 1.
				If bit X of the MSR is cleared to 0, VM entry fails if control X and the "activate tertiary controls" primary processor-based VM-execution control are both 1.
		0		LOADIWKEY
				This control determines whether executions of LOADIWKEY cause VM exits.
		63:1		Reserved
601H	1537	MSR_VR_CURRENT_CONFIG	Package	Power Limit 4 (PL4)
				Package-level maximum power limit (in Watts).
				It is a proactive, instantaneous limit.
		12:0		PL4 Value
				PL4 value in 0.125 A increments. This field is locked by VR_CURRENT_CONFIG[LOCK]. When the LOCK bit is set to 1b, this field becomes Read Only.
		30:13		Reserved
		31		Lock Indication (LOCK)
				This bit will lock the CURRENT_LIMIT settings in this register and will also lock this setting. This means that once set to 1b, the CURRENT_LIMIT setting and this bit become Read Only until the next Warm Reset.
		62:32		Not in use.
		63		Reserved
6A0H	1696	IA32_U_CET		Configure User Mode CET (R/W) See Table 2-2.
6A2H	1698	IA32_S_CET		Configure Supervisor Mode CET (R/W) See Table 2-2.
6A4H	1700	IA32_PL0_SSP		Linear address to be loaded into SSP on transition to privilege level 0. (R/W) See Table 2-2.
6A5H	1701	IA32_PL1_SSP		Linear address to be loaded into SSP on transition to privilege level 1. (R/W) See Table 2-2.
6A6H	1702	IA32_PL2_SSP		Linear address to be loaded into SSP on transition to privilege level 2. (R/W) See Table 2-2.

Table 2-45. Additional MSRs Supported by 11th Generation Intel® Core™ Processors Based on Tiger Lake Microarchitecture

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
6A7H	1703	IA32_PL3_SSP		Linear address to be loaded into SSP on transition to privilege level 3. (R/W) See Table 2-2.
6A8H	1704	IA32_INTERRUPT_SSP_TABLE_ADDR		Linear address of a table of seven shadow stack pointers that are selected in IA-32e mode using the IST index (when not 0) from the interrupt gate descriptor. (R/W) See Table 2-2.
981H	2433	IA32_TME_CAPABILITY		See Table 2-2.
982H	2434	IA32_TME_ACTIVATE		See Table 2-2.
983H	2435	IA32_TME_EXCLUDE_MASK		See Table 2-2.
984H	2436	IA32_TME_EXCLUDE_BASE		See Table 2-2.
990H	2448	IA32_COPY_STATUS ¹	Thread	See Table 2-2.
991H	2449	IA32_IWKEYBACKUP_STATUS ¹	Platform	See Table 2-2.
C82H	3202	IA32_L2_QOS_CFG	Соге	IA32_CR_L2_QOS_CFG
				This MSR provides software an enumeration of the parameters that L2 QoS (Intel RDT) support in any particular implementation.
		0		CDP_ENABLE
				When set to 1, it will enable the code and data prioritization for the L2 CAT/Intel RDT feature.
				When set to 0, code and data prioritization is disabled for L2 CAT/Intel RDT. See Chapter 18, "Debug, Branch Profile, TSC, and Intel® Resource Director Technology (Intel® RDT) Features" for further details on CDP.
		31:1		Reserved
D10H	3220	IA32_L2_QOS_MASK_[0-7]	Package	IA32_CR_L2_QOS_MASK_[0-7]
- D17H	- 3351			Controls MLC (L2) Intel RDT allocation. For more details on CAT/RDT, see Chapter 18, "Debug, Branch Profile, TSC, and Intel® Resource Director Technology (Intel® RDT) Features".
		19:0		WAYS_MASK
				Setting a 1 in this bit X allows threads with CLOS <n> (where N is [0-7]) to allocate to way X in the MLC. Ones are only allowed to be written to ways that physically exist in the MLC (CPUID.4.2:EBX[31:22] will indicate this).</n>
				Writing a 1 to a value beyond the highest way or a non-contiguous set of 1s will cause a #GP on the WRMSR to this MSR.
		31:20		Reserved
D91H	3473	IA32_COPY_LOCAL_TO_PLATFORM ¹	Thread	See Table 2-2.
D92H	3474	IA32_COPY_PLATFORM_TO_LOCAL ¹	Thread	See Table 2-2.

NOTES:

 Further details on Key Locker and usage of this MSR can be found here: https://software.intel.com/content/www/us/en/develop/download/intel-key-locker-specification.html.

2.17.5 MSRs Introduced in 12th and 13th Generation Intel® Core™ Processors Supporting Performance Hybrid Architecture

Table 2-46 lists additional MSRs for 12th and 13th generation Intel Core processors with CPUID DisplayFamily_DisplayModel signatures of 06_97H, 06_9AH, 06_BAH, 06_B7H, and 06_BFH. Table 2-47 lists the MSRs unique to the processor P-core. Table 2-48 lists the MSRs unique to the processor E-core.

The MSRs listed in Table 2-44¹ and Table 2-45 are also supported by these processors. For an MSR listed in Table 2-46, Table 2-47, or Table 2-48 that also appears in the model-specific tables of prior generations, Table 2-46, Table 2-47, and Table 2-48 supersede prior generation tables.

Table 2-46. Additional MSRs Supported by 12th and 13th Generation Intel® Core™ Processors Supporting Performance Hybrid Architecture

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
33H	51	MSR_MEMORY_CTRL	Core	Memory Control Register
		26:0		Reserved.
		27		UC_STORE_THROTTLE
				If set to 1, when enabled, the processor will only allow one in-progress UC store at a time.
		28		UC_LOCK_DISABLE
				If set to 1, a UC load lock will trigger a #GP fault.
		29		Enable #AC(0) exception for split locked accesses:
				Cause #AC(0) exception for split locked access at all CPL irrespective of CR0.AM or EFLAGS.AC.
		30		Reserved.
		31		Reserved.
BCH	188	IA32_MISC_PACKAGE_CTLS	Package	Power Filtering Control (R/W)
				IA32_ARCH_CAPABILITIES[bit 10] enumerates
				support for this MSR. See Table 2-2.
C7H	199	IA32 PMC6	Соге	General Performance Counter 6 (R/W)
C/11	133	1//32_11160	Core	See Table 2-2.
C8H	200	IA32_PMC7	Соге	General Performance Counter 7 (R/W)
				See Table 2-2.
CFH	207	IA32_CORE_CAPABILITIES	Package	IA32 Core Capabilities Register (R/O)
				If CPUID.(EAX=07H, ECX=0):EDX[30] = 1.
				This MSR provides an architectural enumeration function for model-specific behavior.

^{1.} MSRs at the following addresses are not supported in the 12th and 13th generation Intel Core processor E-core: 30CH, 329H, 541H, and 657H. The MSR at address 657H is not supported in the 12th and 13th generation Intel Core processor P-core.

Table 2-46. Additional MSRs Supported by 12th and 13th Generation Intel® Core™ Processors Supporting Performance Hybrid Architecture

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		0		STLB_QOS_SUPPORTED When set to 1, the STLB QoS feature is supported and the STLB QoS MSRs (1A8FH -1A97H) are accessible. When set to 0, access to these MSRs will #GP.
		1		Reserved
		2		FUSA_SUPPORTED
		3		RSM_IN_CPLO_ONLY
				When set to 1, the RSM instruction is only allowed in CPLO (#GP triggered in any CPL != 0).
				When set to 0, then any CPL may execute the RSM instruction.
		4		UC_LOCK_DISABLE_SUPPORTED
				When set to 1, processor supports UC load lock disable feature.
		5		SPLIT_LOCK_DISABLE_SUPPORTED
				When set to 1, the ability to set MEMORY_CONTROL (MSR 33H) bit 29 enables an #AC to be created when a split lock is detected.
		6		SNOOP_FILTER_QOS_SUPPORTED
				When set to 1, the Snoop Filter Qos Mask MSRs are supported.
				When set to 0, access to these MSRs will #GP.
		7		UC_STORE_THROTTLING_SUPPORTED
				When set 1, UC Store throttle capability exist through MSR_MEMORY_CTRL (33H) bit 27.
		31:8		Reserved
E1H	225	IA32_UMWAIT_CONTROL		UMWAIT Control (R/W)
				See Table 2-2.
10AH	266	IA32_ARCH_CAPABILITIES		Enumeration of Architectural Features (R/O) See Table 2-2.
18CH	396	IA32_PERFEVTSEL6	Core	See Table 2-20.
18DH	397	IA32_PERFEVTSEL7	Core	See Table 2-20.
195H	405	IA32_OVERCLOCKING_STATUS	Package	Overclocking Status (R/O) IA32_ARCH_CAPABILITIES[bit 23] enumerates support for this MSR. See Table 2-2.

Table 2-46. Additional MSRs Supported by 12th and 13th Generation Intel® Core™ Processors Supporting Performance Hybrid Architecture

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
1ADH	429	MSR_PRIMARY_TURBO_RATIO_LIMIT	Package	Primary Maximum Turbo Ratio Limit (R/W) Software can configure these limits when MSR_PLATFORM_INFO[28] = 1. Specifies Maximum Ratio Limit for each group.
				Maximum ratio for groups with more cores must decrease monotonically.
		7:0		MAX_TURBO_GROUP_0:
				Maximum turbo ratio limit with 1 core active.
		15:8		MAX_TURBO_GROUP_1:
				Maximum turbo ratio limit with 2 cores active.
		23:16		MAX_TURBO_GROUP_2:
				Maximum turbo ratio limit with 3 cores active.
		31:24		MAX_TURBO_GROUP_3:
				Maximum turbo ratio limit with 4 cores active.
		39:32		MAX_TURBO_GROUP_4:
				Maximum turbo ratio limit with 5 cores active.
		47:40		MAX_TURBO_GROUP_5:
				Maximum turbo ratio limit with 6 cores active.
		55:48		MAX_TURBO_GROUP_6:
				Maximum turbo ratio limit with 7 cores active.
		63:56		MAX_TURBO_GROUP_7:
				Maximum turbo ratio limit with 8 cores active.
493H	1171	IA32_VMX_EXIT_CTLS2		See Table 2-2.
4C7H	1223	IA32_A_PMC6		Full Width Writable IA32_PMC6 Alias (R/W) See Table 2-2.
4C8H	1224	IA32_A_PMC7		Full Width Writable IA32_PMC7 Alias (R/W)
				See Table 2-2.
650H	1616	MSR_SECONDARY_TURBO_RATIO_LIMIT	Package	Secondary Maximum Turbo Ratio Limit (R/W)
				Software can configure these limits when MSR_PLATFORM_INFO[28] = 1.
				Specifies Maximum Ratio Limit for each group. Maximum ratio for groups with more cores must decrease monotonically.
		7:0		MAX_TURBO_GROUP_0:
				Maximum turbo ratio limit with 1 core active.
		15:8		MAX_TURBO_GROUP_1:
				Maximum turbo ratio limit with 2 cores active.
		23:16		MAX_TURBO_GROUP_2:
				Maximum turbo ratio limit with 3 cores active.

Table 2-46. Additional MSRs Supported by 12th and 13th Generation Intel® Core™ Processors Supporting Performance Hybrid Architecture

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec]		
		31:24		MAX_TURBO_GROUP_3:
				Maximum turbo ratio limit with 4 cores active.
		39:32		MAX_TURBO_GROUP_4:
				Maximum turbo ratio limit with 5 cores active.
		47:40		MAX_TURBO_GROUP_5:
				Maximum turbo ratio limit with 6 cores active.
		55:48		MAX_TURBO_GROUP_6:
				Maximum turbo ratio limit with 7 cores active.
		63:56		MAX_TURBO_GROUP_7:
				Maximum turbo ratio limit with 8 cores active.
664H	1636	MSR_MC6_RESIDENCY_COUNTER	Module	Module C6 Residency Counter (R/0)
				Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-States.
		63:0		Time that this module is in module-specific C6 states since last reset. Counts at 1 Mhz frequency.
6E1H	1761	IA32_PKRS		Specifies the PK permissions associated with each protection domain for supervisor pages (R/W) See Table 2-2.
776H	1910	IA32_HWP_CTL		See Table 2-2.
981H	2433	IA32_TME_CAPABILITY		Memory Encryption Capability MSR See Table 2-2.
1200H	4608	IA32_LBR_x_INFO		Last Branch Record Entry X Info Register (R/W)
- 121FH	- 4639			See Table 2-2.
14CEH	5326	IA32_LBR_CTL		Last Branch Record Enabling and Configuration
				Register (R/W)
4.4651.1	5007	Linea LDD DCDTU		See Table 2-2.
14CFH	5327	IA32_LBR_DEPTH		Last Branch Record Maximum Stack Depth Register (R/W)
				See Table 2-2.
1500H	5376	IA32_LBR_x_FROM_IP	 	Last Branch Record Entry X Source IP Register (R/W)
-	-			See Table 2-2.
151FH	5407			
1600H -	5632	IA32_LBR_x_TO_IP		Last Branch Record Entry X Destination IP Register (R/W)
161FH	5663			See Table 2-2.
17D2H	6098	IA32_THREAD_FEEDBACK_CHAR		Thread Feedback Characteristics (R/O)
				See Table 2-2.
17D4H	6100	IA32_HW_FEEDBACK_THREAD_CONFIG		Hardware Feedback Thread Configuration (R/W)
				See Table 2-2.

Table 2-46. Additional MSRs Supported by 12th and 13th Generation Intel® Core™ Processors Supporting Performance Hybrid Architecture

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
17DAH	6106	IA32_HRESET_ENABLE		History Reset Enable (R/W)
				See Table 2-2.

The MSRs listed in Table 2-47 are unique to the 12th and 13th generation Intel Core processor P-core. These MSRs are not supported on the processor E-core.

Table 2-47. MSRs Supported by 12th and 13th Generation Intel® Core™ Processor P-core

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
3F7H	1015	MSR_PEBS_FRONTEND	Thread	FrontEnd Precise Event Condition Select (R/W) See Table 2-39.
540H	1344	MSR_THREAD_UARCH_CTL	Thread	Thread Microarchitectural Control (R/W)
		0		WB_MEM_STRM_LD_DISABLE
				Disable streaming behavior for MOVNTDQA loads to WB memory type. If set, these accesses will be treated like regular cacheable loads (Data will be cached).
		63:1		Reserved
541H	1345	MSR_CORE_UARCH_CTL	Соге	Core Microarchitecture Control MSR (R/W)
				See Table 2-44.
D10H	3220	IA32_L2_QOS_MASK_[0-7]	Соге	IA32_CR_L2_QOS_MASK_[0-7]
- D17H	- 3351			If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] \geq 0.
DITH	3331			Controls MLC (L2) Intel RDT allocation. For more details on CAT/RDT, see Chapter 18, "Debug, Branch Profile, TSC, and Intel® Resource Director Technology (Intel® RDT) Features".
		19:0		WAYS_MASK
				Setting a 1 in this bit X allows threads with CLOS <n> (where N is [0-7]) to allocate to way X in the MLC. Ones are only allowed to be written to ways that physically exist in the MLC (CPUID.4.2:EBX[31:22] will indicate this).</n>
				Writing a 1 to a value beyond the highest way or a non-contiguous set of 1s will cause a #GP on the WRMSR to this MSR.
		31:20		Reserved

The MSRs listed in Table 2-48 are unique to the 12th and 13th generation Intel Core processor E-core. These MSRs are not supported on the processor P-core.

Table 2-48. MSRs Supported by 12th and 13th Generation Intel® Core™ Processor E-core

Regi Add	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
D10H	3220	IA32_L2_QOS_MASK_[0-15]	Module	IA32_CR_L2_QOS_MASK_[0-15]
- D1FH	- 3359			If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] \geq 0.
DIFF	2233			Controls MLC (L2) Intel RDT allocation. For more details on CAT/RDT, see Chapter 18, "Debug, Branch Profile, TSC, and Intel® Resource Director Technology (Intel® RDT) Features".
		19:0		WAYS_MASK
				Setting a 1 in this bit X allows threads with CLOS <n> (where N is [0-7]) to allocate to way X in the MLC. Ones are only allowed to be written to ways that physically exist in the MLC (CPUID.4.2:EBX[31:22] will indicate this).</n>
				Writing a 1 to a value beyond the highest way or a non-contiguous set of 1s will cause a #GP on the WRMSR to this MSR.
		31:20		Reserved
1309H	4873	MSR_RELOAD_FIXED_CTRx		Reload value for IA32_FIXED_CTRx (R/W)
- 130BH	- 4875	47:0		Value loaded into IA32_FIXED_CTRx when a PEBS record is generated while PEBS_EN_FIXEDx = 1 and PEBS_OUTPUT = 01B in IA32_PEBS_ENABLE, and FIXED_CTRx is overflowed.
		63:48		Reserved
14C1H	5313	MSR_RELOAD_PMCx	Core	Reload value for IA32_PMCx (R/W)
- 14C6H	- 5318	47:0		Value loaded into IA32_PMCx when a PEBS record is generated while PEBS_EN_PMCx = 1 and PEBS_OUTPUT = 01B in IA32_PEBS_ENABLE, and PMCx is overflowed.
		63:48		Reserved

Table 2-49 lists the MSRs of uncore PMU for Intel processors with CPUID DisplayFamily_DisplayModel signatures of 06_97H , 06_9AH , 06_BAH , 06_B7H , and 06_B7H .

Table 2-49. Uncore PMU MSRs Supported by 12th and 13th Generation Intel® Core™ Processors

Regi Addı		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
396H	918	MSR_UNC_CBO_CONFIG	Package	Uncore C-Box Configuration Information (R/O)
		3:0		Specifies the number of C-Box units with programmable counters (including processor cores and processor graphics).
		63:4		Reserved
2000H	8192	MSR_UNC_CBO_0_PERFEVTSEL0	Package	Uncore C-Box O, Counter O Event Select MSR
2001H	8193	MSR_UNC_CBO_0_PERFEVTSEL1	Package	Uncore C-Box O, Counter 1 Event Select MSR

Table 2-49. Uncore PMU MSRs Supported by 12th and 13th Generation Intel® Core™ Processors

Regi Add	ster	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
2002H	8194	MSR_UNC_CBO_O_PERFCTRO	Package	Uncore C-Box O, Performance Counter O
2003H	8195	MSR_UNC_CBO_0_PERFCTR1	Package	Uncore C-Box O, Performance Counter 1
2008H	8200	MSR_UNC_CBO_1_PERFEVTSEL0	Package	Uncore C-Box 1, Counter 0 Event Select MSR
2009H	8201	MSR_UNC_CBO_1_PERFEVTSEL1	Package	Uncore C-Box 1, Counter 1 Event Select MSR
200AH	8202	MSR_UNC_CBO_1_PERFCTRO	Package	Uncore C-Box 1, Performance Counter 0
200BH	8203	MSR_UNC_CBO_1_PERFCTR1	Package	Uncore C-Box 1, Performance Counter 1
2010H	8208	MSR_UNC_CBO_2_PERFEVTSELO	Package	Uncore C-Box 2, Counter 0 Event Select MSR
2011H	8209	MSR_UNC_CBO_2_PERFEVTSEL1	Package	Uncore C-Box 2, Counter 1 Event Select MSR
2012H	8210	MSR_UNC_CBO_2_PERFCTRO	Package	Uncore C-Box 2, Performance Counter 0
2013H	8211	MSR_UNC_CBO_2_PERFCTR1	Package	Uncore C-Box 2, Performance Counter 1
2018H	8216	MSR_UNC_CBO_3_PERFEVTSELO	Package	Uncore C-Box 3, Counter 0 Event Select MSR
2019H	8217	MSR_UNC_CBO_3_PERFEVTSEL1	Package	Uncore C-Box 3, Counter 1 Event Select MSR
201AH	8218	MSR_UNC_CBO_3_PERFCTRO	Package	Uncore C-Box 3, Performance Counter 0
201BH	8219	MSR_UNC_CBO_3_PERFCTR1	Package	Uncore C-Box 3, Performance Counter 1
2020H	8224	MSR_UNC_CBO_4_PERFEVTSEL0	Package	Uncore C-Box 4, Counter 0 Event Select MSR
2021H	8225	MSR_UNC_CBO_4_PERFEVTSEL1	Package	Uncore C-Box 4, Counter 1 Event Select MSR
2022H	8226	MSR_UNC_CBO_4_PERFCTRO	Package	Uncore C-Box 4, Performance Counter 0
2023H	8227	MSR_UNC_CBO_4_PERFCTR1	Package	Uncore C-Box 4, Performance Counter 1
2028H	8232	MSR_UNC_CBO_5_PERFEVTSELO	Package	Uncore C-Box 5, Counter 0 Event Select MSR
2029H	8233	MSR_UNC_CBO_5_PERFEVTSEL1	Package	Uncore C-Box 5, Counter 1 Event Select MSR
202AH	8234	MSR_UNC_CBO_5_PERFCTR0	Package	Uncore C-Box 5, Performance Counter 0
202BH	8235	MSR_UNC_CBO_5_PERFCTR1	Package	Uncore C-Box 5, Performance Counter 1
2030H	8240	MSR_UNC_CBO_6_PERFEVTSELO	Package	Uncore C-Box 6, Counter 0 Event Select MSR
2031H	8241	MSR_UNC_CBO_6_PERFEVTSEL1	Package	Uncore C-Box 6, Counter 1 Event Select MSR
2032H	8242	MSR_UNC_CBO_6_PERFCTRO	Package	Uncore C-Box 6, Performance Counter 0
2033H	8243	MSR_UNC_CBO_6_PERFCTR1	Package	Uncore C-Box 6, Performance Counter 1
2038H	8248	MSR_UNC_CBO_7_PERFEVTSEL0	Package	Uncore C-Box 7, Counter 0 Event Select MSR
2039H	8249	MSR_UNC_CBO_7_PERFEVTSEL1	Package	Uncore C-Box 7, Counter 1 Event Select MSR
203AH	8250	MSR_UNC_CBO_7_PERFCTR0	Package	Uncore C-Box 7, Performance Counter 0
203BH	8251	MSR_UNC_CBO_7_PERFCTR1	Package	Uncore C-Box 7, Performance Counter 1
2040H	8256	MSR_UNC_CBO_8_PERFEVTSELO	Package	Uncore C-Box 8, Counter 0 Event Select MSR
2041H	8257	MSR_UNC_CBO_8_PERFEVTSEL1	Package	Uncore C-Box 8, Counter 1 Event Select MSR
2042H	8258	MSR_UNC_CBO_8_PERFCTRO	Package	Uncore C-Box 8, Performance Counter 0
2043H	8259	MSR_UNC_CBO_8_PERFCTR1	Package	Uncore C-Box 8, Performance Counter 1
2048H	8264	MSR_UNC_CBO_9_PERFEVTSEL0	Package	Uncore C-Box 9, Counter 0 Event Select MSR
2049H	8265	MSR_UNC_CBO_9_PERFEVTSEL1	Package	Uncore C-Box 9, Counter 1 Event Select MSR
204AH	8266	MSR_UNC_CBO_9_PERFCTR0	Package	Uncore C-Box 9, Performance Counter 0

Table 2-49. Uncore PMU MSRs Supported by 12th and 13th Generation Intel® Core™ Processors

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
204BH	8267	MSR_UNC_CBO_9_PERFCTR1	Package	Uncore C-Box 9, Performance Counter 1
2FD0H	12240	MSR_UNC_ARB_0_PERFEVTSEL0	Package	Uncore Arb Unit 0, Counter 0 Event Select MSR
2FD1H	12241	MSR_UNC_ARB_0_PERFEVTSEL1	Package	Uncore Arb Unit 0, Counter 1 Event Select MSR
2FD2H	12242	MSR_UNC_ARB_0_PERFCTR0	Package	Uncore Arb Unit 0, Performance Counter 0
2FD3H	12243	MSR_UNC_ARB_0_PERFCTR1	Package	Uncore Arb Unit 0, Performance Counter 1
2FD4H	12244	MSR_UNC_ARB_O_PERF_STATUS	Package	Uncore Arb Unit 0, Performance Status
2FD5H	12245	MSR_UNC_ARB_0_PERF_CTRL	Package	Uncore Arb Unit 0, Performance Control
2FD8H	12248	MSR_UNC_ARB_1_PERFEVTSEL0	Package	Uncore Arb Unit 1, Counter 0 Event Select MSR
2FD9H	12249	MSR_UNC_ARB_1_PERFEVTSEL1	Package	Uncore Arb Unit 1, Counter 1 Event Select MSR
2FDAH	12250	MSR_UNC_ARB_1_PERFCTR0	Package	Uncore Arb Unit 1, Performance Counter 0
2FDBH	12251	MSR_UNC_ARB_1_PERFCTR1	Package	Uncore Arb Unit 1, Performance Counter 1
2FDCH	12252	MSR_UNC_ARB_1_PERF_STATUS	Package	Uncore Arb Unit 1, Performance Status
2FDDH	12253	MSR_UNC_ARB_1_PERF_CTRL	Package	Uncore Arb Unit 1, Performance Control
2FDEH	12254	MSR_UNC_PERF_FIXED_CTRL	Package	Uncore Fixed Counter Control (R/W)
		19:0		Reserved
		20		Enable overflow propagation.
		21		Reserved
		22		Enable counting.
		63:23		Reserved
2FDFH	12255	MSR_UNC_PERF_FIXED_CTR	Package	Uncore Fixed Counter
		43:0		Current count.
		63:44		Reserved
2FF0H	12272	MSR_UNC_PERF_GLOBAL_CTRL	Package	Uncore PMU Global Control
		0		Slice 0 select.
		1		Slice 1 select.
		2		Slice 2 select.
		3		Slice 3 select.
		4		Slice 4 select.
		18:5		Reserved
		29		Enable all uncore counters.
		30		Enable wake on PMI.
		31		Enable Freezing counter when overflow.
		63:32		Reserved
2FF2H	12274	MSR_UNC_PERF_GLOBAL_STATUS	Package	Uncore PMU Main Status
		0		Fixed counter overflowed.
		1		An ARB counter overflowed.
		2		Reserved

Table 2-49. Uncore PMU MSRs Supported by 12th and 13th Generation Intel® Core™ Processors

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		3		A CBox counter overflowed (on any slice).
		63:4		Reserved

2.17.6 MSRs Introduced in Intel® Xeon® Scalable Processor Family

The Intel $^{\circledR}$ Xeon $^{\circledR}$ Scalable Processor Family (CPUID DisplayFamily_DisplayModel = 06_55H) supports the MSRs listed in Table 2-50.

Table 2-50. MSRs Supported by Intel® Xeon® Scalable Processor Family with DisplayFamily_DisplayModel 06_55H

Regi Add		Register Name / Bit Fields (Former MSR Name)	Scope	Bit Description
Hex	Dec			
ЗАН	58	IA32_FEATURE_CONTROL	Thread	Control Features in Intel 64 Processor (R/W)
				See Table 2-2.
		0		Lock (R/WL)
		1		Enable VMX Inside SMX Operation (R/WL)
		2		Enable VMX Outside SMX Operation (R/WL)
		14:8		SENTER Local Functions Enables (R/WL)
		15		SENTER Global Functions Enable (R/WL)
		18		SGX Global Functions Enable (R/WL)
		20		LMCE_ENABLED (R/WL)
		63:21		Reserved
4EH	78	IA32_PPIN_CTL	Package	Protected Processor Inventory Number Enable Control
		(MSR_PPIN_CTL)		(R/W)
		0		LockOut (R/WO)
				See Table 2-2.
		1		Enable_PPIN (R/W)
				See Table 2-2.
		63:2		Reserved
4FH	79	IA32_PPIN	Package	Protected Processor Inventory Number (R/O)
		(MSR_PPIN)		
		63:0		Protected Processor Inventory Number (R/O)
				See Table 2-2.
CEH	206	MSR_PLATFORM_INFO	Package	Platform Information
				Contains power management and other model specific features enumeration. See http://biosbits.org.
		7:0		Reserved
		15:8	Package	Maximum Non-Turbo Ratio (R/O)
				See Table 2-26.
		22:16		Reserved.

Table 2-50. MSRs Supported by Intel® Xeon® Scalable Processor Family with DisplayFamily_DisplayModel 06_55H

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
		23	Package	PPIN_CAP (R/O)
				See Table 2-26.
		27:24		Reserved
		28	Package	Programmable Ratio Limit for Turbo Mode (R/O) See Table 2-26.
		29	Package	Programmable TDP Limit for Turbo Mode (R/O) See Table 2-26.
		30	Package	Programmable TJ OFFSET (R/O) See Table 2-26.
		39:31		Reserved
		47:40	Package	Maximum Efficiency Ratio (R/O)
				See Table 2-26.
		63:48		Reserved
E2H	226	MSR_PKG_CST_CONFIG_CONTROL	Соге	C-State Configuration Control (R/W)
				Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-states.
				See http://biosbits.org.
		2:0		Package C-State Limit (R/W)
				Specifies the lowest processor-specific C-state code name (consuming the least power) for the package. The default is set as factory-configured package C-state limit.
				The following C-state code name encodings are supported:
				000b: C0/C1 (no package C-state support)
				001b: C2
				010b: C6 (non-retention)
				011b: C6 (retention)
				111b: No Package C state limits. All C states supported by the processor are available.
		9:3		Reserved
		10		I/O MWAIT Redirection Enable (R/W)
		14:11		Reserved
		15		CFG Lock (R/W0)
		16		Automatic C-State Conversion Enable (R/W)
				If 1, the processor will convert HALT or MWAT(C1) to MWAIT(C6).
		24:17		Reserved
		25		C3 State Auto Demotion Enable (R/W)
		26		C1 State Auto Demotion Enable (R/W)

Table 2-50. MSRs Supported by Intel® Xeon® Scalable Processor Family with DisplayFamily_DisplayModel 06_55H

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
		27		Enable C3 Undemotion (R/W)
		28		Enable C1 Undemotion (R/W)
		29		Package C State Demotion Enable (R/W)
		30		Package C State UnDemotion Enable (R/W)
		63:31		Reserved
179H	377	IA32_MCG_CAP	Thread	Global Machine Check Capability (R/O)
		7:0		Count
		8		MCG_CTL_P
		9		MCG_EXT_P
		10		MCP_CMCI_P
		11		MCG_TES_P
		15:12		Reserved
		23:16		MCG_EXT_CNT
		24		MCG_SER_P
		25		MCG_EM_P
		26		MCG_ELOG_P
		63:27		Reserved
17DH	381	MSR_SMM_MCA_CAP	THREAD	Enhanced SMM Capabilities (SMM-RO)
				Reports SMM capability Enhancement. Accessible only while in SMM.
		57:0		Reserved
		58		SMM_Code_Access_Chk (SMM-RO)
				If set to 1 indicates that the SMM code access restriction is supported and a host-space interface is available to SMM handler.
		59		Long_Flow_Indication (SMM-RO)
				If set to 1 indicates that the SMM long flow indicator is supported and a host-space interface is available to SMM handler.
		63:60		Reserved
19CH	412	IA32_THERM_STATUS	Соге	Thermal Monitor Status (R/W)
				See Table 2-2.
		0		Thermal Status (R/O)
				See Table 2-2.
		1		Thermal Status Log (R/WC0)
				See Table 2-2.
		2		PROTCHOT # or FORCEPR# Status (R/O)
				See Table 2-2.

Table 2-50. MSRs Supported by Intel® Xeon® Scalable Processor Family with DisplayFamily_DisplayModel 06_55H

Regi Addı		Register Name / Bit Fields (Former MSR Name)	Scope	Bit Description
Hex	Dec			
		3		PROTCHOT # or FORCEPR# Log (R/WC0) See Table 2-2.
		4		Critical Temperature Status (R/O) See Table 2-2.
		5		Critical Temperature Status Log (R/WCO) See Table 2-2.
		6		Thermal Threshold #1 Status (R/O) See Table 2-2.
		7		Thermal Threshold #1 Log (R/WC0) See Table 2-2.
		8		Thermal Threshold #2 Status (R/O) See Table 2-2.
		9		Thermal Threshold #2 Log (R/WC0) See Table 2-2.
		10		Power Limitation Status (R/O) See Table 2-2.
		11		Power Limitation Log (R/WCO) See Table 2-2.
		12		Current Limit Status (R/O) See Table 2-2.
		13		Current Limit Log (R/WCO) See Table 2-2.
		14		Cross Domain Limit Status (R/O) See Table 2-2.
		15		Cross Domain Limit Log (R/WCO) See Table 2-2.
		22:16		Digital Readout (R/O) See Table 2-2.
		26:23		Reserved
		30:27		Resolution in Degrees Celsius (R/O) See Table 2-2.
		31		Reading Valid (R/O) See Table 2-2.
		63:32		Reserved
1A2H	418	MSR_TEMPERATURE_TARGET	Package	Temperature Target
		15:0		Reserved
		23:16		Temperature Target (R/O) See Table 2-26.

Table 2-50. MSRs Supported by Intel® Xeon® Scalable Processor Family with DisplayFamily_DisplayModel 06_55H

Regi Add		Register Name / Bit Fields (Former MSR Name)	Scope	Bit Description
Hex	Dec			
		27:24		TCC Activation Offset (R/W)
				See Table 2-26.
		63:28		Reserved
1ADH	429	MSR_TURBO_RATIO_LIMIT	Package	This register defines the ratio limits. RATIO[0:7] must be populated in ascending order. RATIO[i+1] must be less than or equal to RATIO[i]. Entries with RATIO[i] will be ignored. If any of the rules above are broken, the configuration is silently rejected. If the programmed ratio is: • Above the fused ratio for that core count, it will be clipped to the fuse limits (assuming !OC).
		2.		Below the min supported ratio, it will be clipped.
		7:0		RATIO_0
		150		Defines ratio limits.
		15:8		RATIO_1 Defines ratio limits.
		23:16		
		25.10		RATIO_2 Defines ratio limits.
		31:24		RATIO_3
		31.24		Defines ratio limits.
		39:32		RATIO_4
				Defines ratio limits.
		47:40		RATIO_5
				Defines ratio limits.
		55:48		RATIO_6
				Defines ratio limits.
		63:56		RATIO_7
				Defines ratio limits.
1AEH	430	MSR_TURBO_RATIO_LIMIT_CORES	Package	This register defines the active core ranges for each frequency point. NUMCORE[0:7] must be populated in ascending order. NUMCORE[i+1] must be greater than NUMCORE[i]. Entries with NUMCORE[i] == 0 will be ignored. The last valid entry must have NUMCORE >= the number of cores in the SKU. If any of the rules above are broken, the configuration is silently rejected.
		7:0		NUMCORE_0
				Defines the active core ranges for each frequency point.
		15:8		NUMCORE_1
				Defines the active core ranges for each frequency point.
		23:16		NUMCORE_2
				Defines the active core ranges for each frequency point.

Table 2-50. MSRs Supported by Intel® Xeon® Scalable Processor Family with DisplayFamily_DisplayModel 06_55H

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
		31:24		NUMCORE_3
				Defines the active core ranges for each frequency point.
		39:32		NUMCORE_4
				Defines the active core ranges for each frequency point.
		47:40		NUMCORE_5
				Defines the active core ranges for each frequency point.
		55:48		NUMCORE_6
				Defines the active core ranges for each frequency point.
		63:56		NUMCORE_7
				Defines the active core ranges for each frequency point.
280H	640	IA32_MC0_CTL2	Core	See Table 2-2.
281H	641	IA32_MC1_CTL2	Соге	See Table 2-2.
282H	642	IA32_MC2_CTL2	Core	See Table 2-2.
283H	643	IA32_MC3_CTL2	Core	See Table 2-2.
284H	644	IA32_MC4_CTL2	Package	See Table 2-2.
285H	645	IA32_MC5_CTL2	Package	See Table 2-2.
286H	646	IA32_MC6_CTL2	Package	See Table 2-2.
287H	647	IA32_MC7_CTL2	Package	See Table 2-2.
288H	648	IA32_MC8_CTL2	Package	See Table 2-2.
289H	649	IA32_MC9_CTL2	Package	See Table 2-2.
28AH	650	IA32_MC10_CTL2	Package	See Table 2-2.
28BH	651	IA32_MC11_CTL2	Package	See Table 2-2.
28CH	652	IA32_MC12_CTL2	Package	See Table 2-2.
28DH	653	IA32_MC13_CTL2	Package	See Table 2-2.
28EH	654	IA32_MC14_CTL2	Package	See Table 2-2.
28FH	655	IA32_MC15_CTL2	Package	See Table 2-2.
290H	656	IA32_MC16_CTL2	Package	See Table 2-2.
291H	657	IA32_MC17_CTL2	Package	See Table 2-2.
292H	658	IA32_MC18_CTL2	Package	See Table 2-2.
293H	659	IA32_MC19_CTL2	Package	See Table 2-2.
400H	1024	IA32_MCO_CTL	Core	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
401H	1025	IA32_MCO_STATUS	Core	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
402H	1026	IA32_MCO_ADDR	Core	Bank MCO reports MC errors from the IFU module.
403H	1027	IA32_MC0_MISC	Соге	

Table 2-50. MSRs Supported by Intel® Xeon® Scalable Processor Family with DisplayFamily_DisplayModel 06_55H

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
404H	1028	IA32_MC1_CTL	Соге	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
405H	1029	IA32_MC1_STATUS	Соге	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
406H	1030	IA32_MC1_ADDR	Соге	Bank MC1 reports MC errors from the DCU module.
407H	1031	IA32_MC1_MISC	Соге	
408H	1032	IA32_MC2_CTL	Соге	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
409H	1033	IA32_MC2_STATUS	Соге	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
40AH	1034	IA32_MC2_ADDR	Соге	Bank MC2 reports MC errors from the DTLB module.
40BH	1035	IA32_MC2_MISC	Соге	
40CH	1036	IA32_MC3_CTL	Соге	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
40DH	1037	IA32_MC3_STATUS	Соге	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
40EH	1038	IA32_MC3_ADDR	Core	Bank MC3 reports MC errors from the MLC module.
40FH	1039	IA32_MC3_MISC	Соге	
410H	1040	IA32_MC4_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
411H	1041	IA32_MC4_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
412H	1042	IA32_MC4_ADDR	Package	Bank MC4 reports MC errors from the PCU module.
413H	1043	IA32_MC4_MISC	Package	
414H	1044	IA32_MC5_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
415H	1045	IA32_MC5_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
416H	1046	IA32_MC5_ADDR	Package	Bank MC5 reports MC errors from a link interconnect module.
417H	1047	IA32_MC5_MISC	Package	
418H	1048	IA32_MC6_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
419H	1049	IA32_MC6_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.". Bank MC6 reports MC errors from the integrated I/O
41AH	1050	IA32_MC6_ADDR	Package	module.
41BH	1051	IA32_MC6_MISC	Package	
41CH	1052	IA32_MC7_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
41DH	1053	IA32_MC7_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
41EH	1054	IA32_MC7_ADDR	Package	Bank MC7 reports MC errors from the M2M 0.
41FH	1055	IA32_MC7_MISC	Package	
420H	1056	IA32_MC8_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
421H	1057	IA32_MC8_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
422H	1058	IA32_MC8_ADDR	Package	Bank MC8 reports MC errors from the M2M 1.
423H	1059	IA32_MC8_MISC	Package	
424H	1060	IA32_MC9_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
425H	1061	IA32_MC9_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
426H	1062	IA32_MC9_ADDR	Package	Banks MC9 - MC11 report MC errors from the CHA
427H	1063	IA32_MC9_MISC	Package	

Table 2-50. MSRs Supported by Intel® Xeon® Scalable Processor Family with DisplayFamily_DisplayModel 06_55H

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
428H	1064	IA32_MC10_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
429H	1065	IA32_MC10_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
42AH	1066	IA32_MC10_ADDR	Package	Banks MC9 - MC11 report MC errors from the CHA.
42BH	1067	IA32_MC10_MISC	Package	
42CH	1068	IA32_MC11_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
42DH	1069	IA32_MC11_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
42EH	1070	IA32_MC11_ADDR	Package	Banks MC9 - MC11 report MC errors from the CHA.
42FH	1071	IA32_MC11_MISC	Package	
430H	1072	IA32_MC12_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
431H	1073	IA32_MC12_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
432H	1074	IA32_MC12_ADDR	Package	Banks MC12 report MC errors from each channel of a link interconnect module.
433H	1075	IA32_MC12_MISC	Package	
434H	1076	IA32_MC13_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
435H	1077	IA32_MC13_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
436H	1078	IA32_MC13_ADDR	Package	Banks MC13 through MC 18 report MC errors from the integrated memory controllers.
437H	1079	IA32_MC13_MISC	Package	
438H	1080	IA32_MC14_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
439H	1081	IA32_MC14_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
43AH	1082	IA32_MC14_ADDR	Package	Banks MC13 through MC 18 report MC errors from the integrated memory controllers.
43BH	1083	IA32_MC14_MISC	Package	
43CH	1084	IA32_MC15_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
43DH	1085	IA32_MC15_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
43EH	1086	IA32_MC15_ADDR	Package	Banks MC13 through MC 18 report MC errors from the integrated memory controllers.
43FH	1087	IA32_MC15_MISC	Package	
440H	1088	IA32_MC16_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
441H	1089	IA32_MC16_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
442H	1090	IA32_MC16_ADDR	Package	Banks MC13 through MC 18 report MC errors from the integrated memory controllers
443H	1091	IA32_MC16_MISC	Package	
444H	1092	IA32_MC17_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
445H	1093	IA32_MC17_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
446H	1094	IA32_MC17_ADDR	Package	Banks MC13 through MC 18 report MC errors from the integrated memory controllers.
447H	1095	IA32_MC17_MISC	Package	
448H	1096	IA32_MC18_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
449H	1097	IA32_MC18_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
44AH	1098	IA32_MC18_ADDR	Package	Banks MC13 through MC 18 report MC errors from the integrated memory controllers.
44BH	1099	IA32_MC18_MISC	Package	a megrated memory controllers.

Table 2-50. MSRs Supported by Intel® Xeon® Scalable Processor Family with DisplayFamily_DisplayModel 06_55H

	ister ress	Register Name / Bit Fields (Former MSR Name)	Scope	Bit Description
Hex	Dec			
44CH	1100	IA32_MC19_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
44DH	1101	IA32_MC19_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.". Bank MC19 reports MC errors from a link interconnect module.
44EH	1102	IA32_MC19_ADDR	Package	
44FH	1103	IA32_MC19_MISC	Package	- module.
606H	1542	MSR_RAPL_POWER_UNIT	Package	Unit Multipliers Used in RAPL Interfaces (R/O)
		3:0	Package	Power Units
				See Section 15.10.1, "RAPL Interfaces."
		7:4	Package	Reserved
		12:8	Package	Energy Status Units
				Energy related information (in Joules) is based on the multiplier, 1/2^ESU; where ESU is an unsigned integer represented by bits 12:8. Default value is 0EH (or 61 micro-joules).
		15:13	Package	Reserved
		19:16	Package	Time Units
				See Section 15.10.1, "RAPL Interfaces."
		63:20		Reserved
618H	1560	MSR_DRAM_POWER_LIMIT	Package	DRAM RAPL Power Limit Control (R/W)
				See Section 15.10.5, "DRAM RAPL Domain."
619H	1561	MSR_DRAM_ENERGY_STATUS	Package	DRAM Energy Status (R/O)
				Energy consumed by DRAM devices.
		31:0		Energy in 15.3 micro-joules. Requires BIOS configuration to enable DRAM RAPL mode 0 (Direct VR).
		63:32		Reserved
61BH	1563	MSR_DRAM_PERF_STATUS	Package	DRAM Performance Throttling Status (R/O)
				See Section 15.10.5, "DRAM RAPL Domain."
61CH	1564	MSR_DRAM_POWER_INFO	Package	DRAM RAPL Parameters (R/W)
				See Section 15.10.5, "DRAM RAPL Domain."
620H	1568	MSR UNCORE_RATIO_LIMIT	Package	Uncore Ratio Limit (R/W)
				Out of reset, the min_ratio and max_ratio fields represent the widest possible range of uncore frequencies. Writing to these fields allows software to control the minimum and the maximum frequency that hardware will select.
		63:15		Reserved
		14:8		MIN_RATIO
				Writing to this field controls the minimum possible ratio of the LLC/Ring.
		7		Reserved

Table 2-50. MSRs Supported by Intel® Xeon® Scalable Processor Family with DisplayFamily_DisplayModel 06_55H

Register Address		Register Name / Bit Fields	Scope	Bit Description	
Hex	Dec	(Former MSR Name)			
		6:0		MAX_RATIO	
				This field is used to limit the max ratio of the LLC/Ring.	
639H	1593	MSR_PPO_ENERGY_STATUS	Package	Reserved (R/O)	
				Reads return 0.	
C8DH	3213	IA32_QM_EVTSEL	THREAD	Monitoring Event Select Register (R/W) If CPUID.(EAX=07H, ECX=0):EBX.RDT-M[bit 12] = 1.	
		7:0		EventID (R/W)	
				Event encoding:	
				0x00: No monitoring.	
				0x01: L3 occupancy monitoring.	
				0x02: Total memory bandwidth monitoring.	
				0x03: Local memory bandwidth monitoring.	
				All other encoding reserved.	
		31:8		Reserved	
		41:32		RMID (R/W)	
		63:42		Reserved	
C8FH	3215	IA32_PQR_ASSOC	THREAD	Resource Association Register (R/W)	
		9:0		RMID	
		31:10		Reserved	
		51:32		COS (R/W)	
		63: 52		Reserved	
C90H	3216	IA32_L3_QOS_MASK_0	Package	L3 Class Of Service Mask - COS 0 (R/W)	
				If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=0.	
		0:19		CBM: Bit vector of available L3 ways for COS 0 enforcement.	
		63:20		Reserved	
C91H	3217	91H 3217	IA32_L3_QOS_MASK_1	Package	L3 Class Of Service Mask - COS 1 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=1.
		0:19		CBM: Bit vector of available L3 ways for COS 1 enforcement.	
		63:20		Reserved	
C92H	3218	IA32_L3_QOS_MASK_2	Package	L3 Class Of Service Mask - COS 2 (R/W)	
CSETT	3210	"132_c3_q03_;	1 dekage	If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=2.	
		0:19		CBM: Bit vector of available L3 ways for COS 2 enforcement.	
		63:20		Reserved	
C93H	3219	IA32_L3_QOS_MASK_3	Package	L3 Class Of Service Mask - COS 3 (R/W).	
				If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=3.	
		0:19		CBM: Bit vector of available L3 ways for COS 3 enforcement.	

Table 2-50. MSRs Supported by Intel® Xeon® Scalable Processor Family with DisplayFamily_DisplayModel 06_55H

Register Address		Register Name / Bit Fields	Scope	Bit Description	
Hex	Dec	(Former MSR Name)			
		63:20		Reserved	
C94H	3220	IA32_L3_QOS_MASK_4	Package	L3 Class Of Service Mask - COS 4 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=4.	
		0:19		CBM: Bit vector of available L3 ways for COS 4 enforcement.	
		63:20		Reserved	
C95H	3221	IA32_L3_QOS_MASK_5	Package	L3 Class Of Service Mask - COS 5 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=5.	
		0:19		CBM: Bit vector of available L3 ways for COS 5 enforcement.	
		63:20		Reserved	
C96H	3222	IA32_L3_QOS_MASK_6	Package	L3 Class Of Service Mask - COS 6 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=6.	
		0:19		CBM: Bit vector of available L3 ways for COS 6 enforcement.	
		63:20		Reserved	
C97H	3223	IA32_L3_QOS_MASK_7	Package	L3 Class Of Service Mask - COS 7 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=7.	
		0:19		CBM: Bit vector of available L3 ways for COS 7 enforcement.	
		63:20		Reserved	
C98H	3224	IA32_L3_QOS_MASK_8	Package	L3 Class Of Service Mask - COS 8 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=8.	
		0:19		CBM: Bit vector of available L3 ways for COS 8 enforcement.	
		63:20		Reserved	
C99H	3225	IA32_L3_QOS_MASK_9	Package	L3 Class Of Service Mask - COS 9 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=9.	
		0:19		CBM: Bit vector of available L3 ways for COS 9 enforcement.	
		63:20		Reserved	
C9AH	3226	IA32_L3_QOS_MASK_10	Package	L3 Class Of Service Mask - COS 10 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=10.	
		0:19		CBM: Bit vector of available L3 ways for COS 10 enforcement.	
		63:20		Reserved	
C9BH	3227	IA32_L3_QOS_MASK_11	Package	L3 Class Of Service Mask - COS 11 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=11.	

Table 2-50. MSRs Supported by Intel® Xeon® Scalable Processor Family with DisplayFamily_DisplayModel 06_55H

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
		0:19		CBM: Bit vector of available L3 ways for COS 11 enforcement.
		63:20		Reserved
C9CH	3228	IA32_L3_QOS_MASK_12	Package	L3 Class Of Service Mask - COS 12 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=12.
		0:19		CBM: Bit vector of available L3 ways for COS 12 enforcement.
		63:20		Reserved
C9DH	3229	IA32_L3_QOS_MASK_13	Package	L3 Class Of Service Mask - COS 13 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=13.
		0:19		CBM: Bit vector of available L3 ways for COS 13 enforcement.
		63:20		Reserved
C9EH	3230	IA32_L3_QOS_MASK_14	Package	L3 Class Of Service Mask - COS 14 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=14.
		0:19		CBM: Bit vector of available L3 ways for COS 14 enforcement.
		63:20		Reserved
C9FH	3231	IA32_L3_QOS_MASK_15	Package	L3 Class Of Service Mask - COS 15 (R/W) If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0] >=15.
		0:19		CBM: Bit vector of available L3 ways for COS 15 enforcement.
		63:20		Reserved

2.17.7 MSRs Specific to 3rd Generation Intel® Xeon® Scalable Processor Family based on Ice Lake Microarchitecture

The 3rd generation Intel[®] Xeon[®] Scalable Processor Family based on Ice Lake microarchitecture (CPUID DisplayFamily_DisplayModel signatures of 06_6AH and 06_6CH) support the MSRs listed in Table 2-51.

Table 2-51. MSRs Supported by 3rd Generation Intel® Xeon® Scalable Processor Family with DisplayFamily_DisplayModel Signatures of 06_6AH and 06_6CH

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
612H	1554	MSR_PACKAGE_ENERGY_TIME_STATUS	Package	Package energy consumed by the entire CPU (R/W)
		31:0		Total amount of energy consumed since last reset.

Table 2-51. MSRs Supported by 3rd Generation Intel® Xeon® Scalable Processor Family with DisplayFamily_DisplayModel Signatures of 06_6AH and 06_6CH

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		63:32		Total time elapsed when the energy was last updated. This is a monotonic increment counter with auto wrap back to zero after overflow. Unit is 10ns.
618H	1560	MSR_DRAM_POWER_LIMIT	Package	Allows software to set power limits for the DRAM domain and measurement attributes associated with each limit.
		14:0		DRAM_PP_PWR_LIM:
				Power Limit[0] for DDR domain. Units = Watts, Format = 11.3, Resolution = 0.125W, Range = 0-2047.875W.
		15		PWR_LIM_CTRL_EN:
				Power Limit[0] enable bit for DDR domain.
		16		Reserved
		23:17		CTRL_TIME_WIN:
				Power Limit[0] time window Y value, for DDR domain. Actual time_window for RAPL is:
				(1/1024 seconds) * (1+(x/4)) * (2^y)
		62:24		Reserved
		63		PP_PWR_LIM_LOCK:
				When set, this entire register becomes read-only. This bit will typically be set by BIOS during boot.
619H	1561	MSR_DRAM_ENERGY_STATUS	Package	DRAM Energy Status (R/O)
				See Section 15.10.5, "DRAM RAPL Domain."
		31:0		Energy in 15.3 micro-joules. Requires BIOS configuration to enable DRAM RAPL mode 0 (Direct VR).
		63:32		Reserved
61BH	1563	MSR_DRAM_PERF_STATUS	Package	DRAM Performance Throttling Status (R/O)
				See Section 15.10.5, "DRAM RAPL Domain."
61CH	1564	MSR_DRAM_POWER_INFO	Package	DRAM Power Parameters (R/W)
		14:0		Spec DRAM Power (DRAM_TDP):
				The Spec power allowed for DRAM. The TDP setting is
				typical (not guaranteed).
				The units for this value are defined in
				MSR_DRAM_POWER_INFO_UNIT[PWR_UNIT].
		15		Reserved
		30:16		Minimal DRAM Power (DRAM_MIN_PWR):
				The minimal power setting allowed for DRAM. Lower
				values will be clamped to this value. The minimum
				setting is typical (not guaranteed).
				The units for this value are defined in MSR_DRAM_POWER_INFO_UNIT[PWR_UNIT].
				1 131 UNAITE OWER INFO ON TEME ON THE

Table 2-51. MSRs Supported by 3rd Generation Intel® Xeon® Scalable Processor Family with DisplayFamily_DisplayModel Signatures of 06_6AH and 06_6CH

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		31		Reserved
		46:32		Maximal Package Power (DRAM_MAX_PWR): The maximal power setting allowed for DRAM. Higher values will be clamped to this value. The maximum setting is typical (not guaranteed). The units for this value are defined in MSR_DRAM_POWER_INFO_UNIT[PWR_UNIT].
		47		Reserved
		54:48		Maximal Time Window (DRAM_MAX_WIN): The maximal time window allowed for the DRAM. Higher values will be clamped to this value. x = PKG_MAX_WIN[54:53] y = PKG_MAX_WIN[52:48] The timing interval window is a floating-point number given by 1.x *power(2,y). The unit of measurement is defined in MSR_DRAM_POWER_INFO_UNIT[TIME_UNIT].
		62:55		Reserved
		63		LOCK: Lock bit to lock the register.
981H	2433	IA32_TME_CAPABILITY		See Table 2-2.
982H	2434	IA32_TME_ACTIVATE		See Table 2-2.
983H	2435	IA32_TME_EXCLUDE_MASK		See Table 2-2.
984H	2436	IA32_TME_EXCLUDE_BASE		See Table 2-2.

2.17.8 MSRs Specific to 4th Generation Intel® Xeon® Scalable Processor Family Based on Sapphire Rapids Microarchitecture

The 4th generation Intel[®] Xeon[®] Scalable Processor Family based on Sapphire Rapids microarchitecture (CPUID DisplayFamily_DisplayModel signature of 06_8FH) supports the MSRs listed in Section 2.17, "MSRs In the 6th Generation, 7th Generation, 8th Generation, 9th Generation, 10th Generation, 11th Generation, 12th Generation, and 13th Generation Intel® Core[™] Processors, Intel® Xeon® Scalable Processor Family, 2nd, 3rd, and 4th Generation Intel® Xeon® Scalable Processors, and Intel® Xeon® E processors," including Table 2-52. For an MSR listed in Table 2-52 that also appears in the model-specific tables of prior generations, Table 2-52 supersedes prior generation tables.

Table 2-52. Additional MSRs Supported by the 4th Generation Intel® Xeon® Scalable Processor Family with a DisplayFamily_DisplayModel Signature of 06_8FH

	ister Iress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
A7H	167	MSR_BIOS_DEBUG	Thread	BIOS DEBUG (R/O) See Table 2-45.
BCH	188	IA32_MISC_PACKAGE_CTLS	Package	Power Filtering Control (R/W) IA32_ARCH_CAPABILITIES[bit 10] enumerates support for this MSR. See Table 2-2.
CFH	207	IA32_CORE_CAPABILITY	Core	IA32 Core Capabilities Register (R/W) If CPUID.(EAX=07H, ECX=0):EDX[30] = 1. This MSR provides an architectural enumeration function for model-specific behavior.
		0		Reserved: returns zero.
		1		Reserved: returns zero.
		2		INTEGRITY_CAPABILITIES
				When set to 1, the processor supports MSR_INTEGRITY_CAPABILITIES.
		3		RSM_IN_CPLO_ONLY
				Indicates that RSM will only be allowed in CPLO and will #GP for all non-CPLO privilege levels.
		4		UC_LOCK_DISABLE_SUPPORTED
				When set to 1, processor supports UC load lock disable feature.
		5		SPLIT_LOCK_DISABLE_SUPPORTED
				Indicates that there is core support for #AC on split lock detection feature.
		6		Reserved: returns zero.
		7		UC_STORE_THROTTLING_SUPPORTED
				Indicates that the snoop filter quality of service MSRs are supported on this core. This is based on the existence of a non-inclusive cache and the L2/MLC QoS feature supported.
		63:8		Reserved: returns zero.
E1H	225	IA32_UMWAIT_CONTROL		UMWAIT Control (R/W)
				See Table 2-2.
EDH	237	MSR_RAR_CONTROL	Thread	RAR Control (R/W)
		63:32		Reserved.
		31		ENABLE
				RAR events are recognized. When RAR is not enabled, RARs are dropped.

Table 2-52. Additional MSRs Supported by the 4th Generation Intel® Xeon® Scalable Processor Family with a DisplayFamily_DisplayModel Signature of 06_8FH

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		30		IGNORE_IF Allow RAR servicing at the RLP regardless of the value of RFLAGS.IF.
		29:0		Reserved.
EEH	238	MSR_RAR_ACTION_VECTOR_BASE	Thread	Pointer to RAR Action Vector (R/W)
		63:MAXPHYADDR		Reserved.
		MAXPHYADDR-1:6		VECTOR_PHYSICAL_ADDRESS
				Pointer to the physical address of the 64B aligned RAR action vector.
		5:0		Reserved.
EFH	239	MSR_RAR_PAYLOAD_TABLE_BASE	Thread	Pointer to Base of RAR Payload Table (R/W)
		63:MAXPHYADDR		Reserved.
		MAXPHYADDR-1:12		TABLE_PHYSICAL_ADDRESS
				Pointer to the base physical address of the 4K aligned RAR payload table.
		11:0		Reserved.
F0H	240	MSR_RAR_INFO	Thread	Read Only RAR Information (RO)
		63:38		Always zero.
		37:32		Table Max Index
				Maximum supported payload table index.
		31:0		Supported payload type bitmap. A value of 1 in bit position [i] indicates that payload type [i] is supported.
105H	261	MSR_CORE_BIST	Соге	Core BIST (R/W)
				Controls Array BIST activation and status checking as part of FUSA.
		31:0		BIST_ARRAY
				Bitmap indicating which arrays to run BIST on (WRITE).
				Bitmap indicating which arrays were not processed, i.e., completion mask (READ).
		39:32		BANK
				Array bank of the [least significant set bit] array indicated in EAX to start BIST(WRITE).
				Array bank interrupted or failed (READ).
		47:40		DWORD
				Array dword of the [least significant set bit] array indicated in EAX to start BIST (WRITE).
				Array dword interrupted or failed (READ).
		62:48		Reserved

Table 2-52. Additional MSRs Supported by the 4th Generation Intel® Xeon® Scalable Processor Family with a DisplayFamily_DisplayModel Signature of 06_8FH

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec]		
		63		CTRL_RESULT Indicates whether WRMSR should signal Machine- Check upon BIST-error (WRITE). BIST result PASS(0)/FAIL(1) of the (least significant set bit) array indicated in EAX (READ).
10AH	266	IA32_ARCH_CAPABILITIES		Enumeration of Architectural Features (R/O) See Table 2-2.
1A4H	420	MSR_PREFETCH_CONTROL		Prefetch Disable Bits (R/W)
		0		L2_HARDWARE_PREFETCHER_DISABLE
				If 1, disables the L2 hardware prefetcher, which fetches additional lines of code or data into the L2 cache.
		1		L2_ADJACENT_CACHE_LINE_PREFETCHER_DISABLE
				If 1, disables the adjacent cache line prefetcher, which fetches the cache line that comprises a cache line pair (128 bytes).
		2		DCU_HARDWARE_PREFETCHER_DISABLE
				If 1, disables the L1 data cache prefetcher, which fetches the next cache line into L1 data cache.
		3		DCU_IP_PREFETCHER_DISABLE
				If 1, disables the L1 data cache IP prefetcher, which uses sequential load history (based on instruction pointer of previous loads) to determine whether to prefetch additional lines.
		4		Reserved.
		5		AMP_PREFETCH_DISABLE
				If 1, disables the L2 Adaptive Multipath Probability (AMP) prefetcher.
		63:6		Reserved.
1ADH	429	MSR_PRIMARY_TURBO_RATIO_LIMIT	Package	Primary Maximum Turbo Ratio Limit (R/W) See Table 2-46.
1AEH	430	MSR_TURBO_RATIO_LIMIT_CORES	Package	See Table 2-50.
1C4H	452	IA32_XFD		Extended Feature Detect (R/W) See Table 2-2.
1C5H	453	IA32_XFD_ERR		XFD Error Code (R/W)
10311	733	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,		See Table 2-2.
2C2H	706	MSR_COPY_SCAN_HASHES	Die	COPY_SCAN_HASHES (W)
	_	63:0		SCAN_HASH_ADDR
				Contains the linear address of the SCAN Test HASH Binary loaded into memory.
2C3H	707	MSR_SCAN_HASHES_STATUS		SCAN_HASHES_STATUS (R/O)

Table 2-52. Additional MSRs Supported by the 4th Generation Intel® Xeon® Scalable Processor Family with a DisplayFamily_DisplayModel Signature of 06_8FH

Regi Addı		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		15:0	Die	CHUNK_SIZE
				Chunk size of the test in KB.
		23:16	Die	NUM_CHUNKS
				Total number of chunks.
		31:24		Reserved: all zeros.
		39:32	Thread	ERROR_CODE
				The error-code refers to the LP that runs WRMSR(2C2H).
				0x0: No error reported.
				0x1: Attempt to copy scan-hashes when copy already in progress.
				0x2: Secure Memory not set up correctly.
				0x3: Scan-image header Image_info.ProgramID doesn't match RDMSR(2D9H)[31:24], or scan-image header Processor-Signature doesn't match F/M/S, or scan-image header Processor-Flags doesn't match PlatformID.
				0x4: Reserved
				0x5: Integrity check failed.
				0x6: Re-install of scan test image attempted when current scan test image is in use by other LPs.
		50:40		Reserved: set to all zeros.
		62:51	Die	MAX_CORE_LIMIT
				Maximum Number of cores that can run Intel® In-field Scan simultaneously minus 1.
				0 means 1 core at a time.
		63	Die	Valid
				Valid bit is set when COPY_SCAN_HASHES has completed successfully.
2C4H	708	MSR_AUTHENTICATE_AND_COPY_CHUNK	Die	AUTHENTICATE_AND_COPY_CHUNK (W)
		7:0		CHUNK_INDEX
				Chunk Index, should be less than the total number of chunks defined by NUM_CHUNKS (MSR_SCAN_HASHES_STATUS[23:16]).
		63:8		CHUNK_ADDR
				Bits 63:8 of 256B aligned Linear address of scan chunk in memory.
2C5H	709	MSR_CHUNKS_AUTHENTICATION_STATUS		CHUNKS_AUTHENTICATION_STATUS (R/O)
		7:0	Die	VALID_CHUNKS
				Total number of Valid (authenticated) chunks.
		15:8	Die	TOTAL_CHUNKS
				Total number of chunks.

Table 2-52. Additional MSRs Supported by the 4th Generation Intel® Xeon® Scalable Processor Family with a DisplayFamily_DisplayModel Signature of 06_8FH

Regi Addı		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		31:16		Reserved: all zeros.
		39:32	Thread	ERROR_CODE
				The error code refers to the LP that runs WRMSR(2C4H).
				0x0: No error reported.
				Ox1: Attempt to authenticate a CHUNK which is already marked as authentic or is currently being installed by another core.
				0x2: CHUNK authentication error. HASH of chunk did not match expected value.
		63:40		Reserved: set to all zeros.
2C6H	710	MSR_ACTIVATE_SCAN	Thread	ACTIVATE_SCAN (W)
		7:0		CHUNK_START_INDEX
				Indicates chunk index to start from.
		15:8		CHUNK_STOP_INDEX
				Indicates what chunk index to stop at (inclusive).
		31:16		Reserved: all zeros.
		62:32		THREAD_WAIT_DELAY
				TSC based delay to allow threads to rendezvous.
		63		SIGNAL_MCE
				If 1, then on scan-error log MC in MC4_STATUS and signal MCE if machine check signaling enabled in MC4_CTL[0].
				If 0, then no logging/no signaling.
2C7H	711	MSR_SCAN_STATUS		SCAN_STATUS (R/O)
		7:0	Core	CHUNK_NUM
				SCAN Chunk that was reached.
		15:8	Core	CHUNK_STOP_INDEX
				Indicates what chunk index to stop at (inclusive). Maps to same field in WRMSR(ACTIVATE_SCAN).
		31:16		Reserved: return all zeros.

Table 2-52. Additional MSRs Supported by the 4th Generation Intel® Xeon® Scalable Processor Family with a DisplayFamily_DisplayModel Signature of 06_8FH

Regi Add	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		39:32	Thread	ERROR_CODE
				0x0: No error.
				Ox1: SCAN operation did not start. Other thread did not join in time.
				0x2: SCAN operation did not start. Interrupt occurred prior to threads rendezvous.
				0x3: SCAN operation did not start. Power Management conditions are inadequate to run Intel Infield Scan.
				Ox4: SCAN operation did not start. Non-valid chunks in the range CHUNK_STOP_INDEX : CHUNK_START_INDEX.
				0x5: SCAN operation did not start. Mismatch in arguments between threads T0/T1.
				Ox6: SCAN operation did not start. Core not capable of performing SCAN currently.
				Ox8: SCAN operation did not start. Exceeded number of Logical Processors (LP) allowed to run Intel In-field Scan concurrently. MAX_CORE_LIMIT exceeded.
				Ox9: Interrupt occurred. Scan operation aborted prematurely, not all chunks requested have been executed.
		61:40		Reserved: return all zeros.
		62	Соге	SCAN_CONTROL_ERROR
				Scan-System-Controller malfunction.
		63	Соге	SCAN_SIGNATURE_ERROR
				Core failed SCAN-SIGNATURE checking for this chunk.
2C8H	712	MSR_SCAN_MODULE_ID	Module	SCAN_MODULE_ID (R/O)
		31:0		RevID of the currently installed scan test image. Maps to Revision field in external header (offset 4).
		63:32		Reserved: return all zeros.
2C9H	713	MSR_LAST_SAF_WP	Core	LAST_SAF_WP (R/O)
		31:0		LAST_WP
				Provides information about the core when the last WRMSR(ACTIVATE_SCAN) was executed. Available only if enumerated in MSR_INTEGRITY_CAPABILITIES[10:9].
		63:32		Reserved: return all zeros.
2D9H	729	MSR_INTEGRITY_CAPABILITIES	Module	INTEGRITY_CAPABILITIES (R/O)
		0		STARTUP_SCAN_BIST
				When set, supports Intel In-field Scan.
		3:1		Reserved: return all zeros.

Table 2-52. Additional MSRs Supported by the 4th Generation Intel® Xeon® Scalable Processor Family with a DisplayFamily_DisplayModel Signature of 06_8FH

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
		4		PERIODIC_SCAN_BIST
				When set, supports Intel In-field Scan.
		23:5		Reserved: return all zeros.
		31:24		ID of the scan programs supported for this part. WRMSR(2C2H) verifies this value against the corresponding value in the scan-image header, i.e., Image_info.
410H	1040	IA32_MC4_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs" through
411H	1041	IA32_MC4_STATUS	Package	Section 16.3.2.4, "IA32_MCi_MISC MSRs.".
412H	1042	IA32_MC4_ADDR	Package	Bank MC4 reports MC errors from the PCU module. — If SIGNAL_MCE is set, a Scan Status is logged in
413H	1043	IA32_MC4_MISC	Package	MC4_STATUS and MC4_MISC.
492H	1170	IA32_VMX_PROCBASED_CTLS3		Capability Reporting Register of Tertiary Processor- Based VM-Execution Controls (R/O) See Table 2-2.
493H	1171	IA32_VMX_EXIT_CTLS2		Capability Reporting Register of Secondary VM-Exit Controls (R/O) See Table 2-2.
540H	1344	MSR_THREAD_UARCH_CTL	Thread	Thread Microarchitectural Control (R/W) See Table 2-47.
65CH	1628	MSR_PLATFORM_POWER_LIMIT	Platform	Platform Power Limit Control (R/W-L) See Table 2-39.
6A0H	1696	IA32_U_CET		Configure User Mode CET (R/W) See Table 2-2.
6A2H	1698	IA32_S_CET		Configure Supervisor Mode CET (R/W) See Table 2-2.
6A4H	1700	IA32_PL0_SSP		Linear address to be loaded into SSP on transition to privilege level 0. (R/W) See Table 2-2.
6A5H	1701	IA32_PL1_SSP		Linear address to be loaded into SSP on transition to privilege level 1. (R/W) See Table 2-2.
6A6H	1702	IA32_PL2_SSP		Linear address to be loaded into SSP on transition to privilege level 2. (R/W) See Table 2-2.
6A7H	1703	IA32_PL3_SSP		Linear address to be loaded into SSP on transition to privilege level 3. (R/W) See Table 2-2.

Table 2-52. Additional MSRs Supported by the 4th Generation Intel® Xeon® Scalable Processor Family with a DisplayFamily_DisplayModel Signature of 06_8FH

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
6A8H	1704	IA32_INTERRUPT_SSP_TABLE_ADDR		Linear address of a table of seven shadow stack pointers that are selected in IA-32e mode using the IST index (when not 0) from the interrupt gate descriptor. (R/W) See Table 2-2.
6E1H	1761	IA32_PKRS		Specifies the PK permissions associated with each protection domain for supervisor pages (R/W) See Table 2-2.
776H	1910	IA32_HWP_CTL		See Table 2-2.
981H	2433	IA32_TME_CAPABILITY		Memory Encryption Capability MSR See Table 2-2.
985H	2437	IA32_UINTR_RR		User Interrupt Request Register (R/W) See Table 2-2.
986H	2438	IA32_UINTR_HANDLER		User Interrupt Handler Address (R/W) See Table 2-2.
987H	2439	IA32_UINTR_STACKADJUST		User Interrupt Stack Adjustment (R/W) See Table 2-2.
988H	2440	IA32_UINTR_MISC		User-Interrupt Target-Table Size and Notification Vector (R/W) See Table 2-2.
989H	2441	IA32_UINTR_PD		User Interrupt PID Address (R/W) See Table 2-2.
98AH	2442	IA32_UINTR_TT		User-Interrupt Target Table (R/W) See Table 2-2.
C70H	3184	MSR_B1_PMON_EVNT_SEL0	Package	Uncore B-box 1 perfmon event select MSR.
C71H	3185	MSR_B1_PMON_CTR0	Package	Uncore B-box 1 perfmon counter MSR.
C72H	3186	MSR_B1_PMON_EVNT_SEL1	Package	Uncore B-box 1 perfmon event select MSR.
C73H	3187	MSR_B1_PMON_CTR1	Package	Uncore B-box 1 perfmon counter MSR.
C74H	3188	MSR_B1_PMON_EVNT_SEL2	Package	Uncore B-box 1 perfmon event select MSR.
C75H	3189	MSR_B1_PMON_CTR2	Package	Uncore B-box 1 perfmon counter MSR.
C76H	3190	MSR_B1_PMON_EVNT_SEL3	Package	Uncore B-box 1vperfmon event select MSR.
C77H	3191	MSR_B1_PMON_CTR3	Package	Uncore B-box 1 perfmon counter MSR.
C82H	3122	MSR_W_PMON_BOX_OVF_CTRL	Package	Uncore W-box perfmon local box overflow control MSR.
C8FH	3215	IA32_PQR_ASSOC		See Table 2-2.

Table 2-52. Additional MSRs Supported by the 4th Generation Intel® Xeon® Scalable Processor Family with a DisplayFamily_DisplayModel Signature of 06_8FH

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
C90H - C9EH	3216 - 3230	IA32_L3_QOS_MASK_0 through IA32_L3_QOS_MASK_14	Package	See Table 2-50.
D10H - D17H	3344 - 3351	IA32_L2_QOS_MASK_[0-7]	Core	$\begin{split} & \text{IA32_CR_L2_QOS_MASK_[0-7]} \\ & \text{If CPUID.(EAX=10H, ECX=1):EDX.COS_MAX[15:0]} \geq 0. \\ & \text{See Table 2-2.} \end{split}$
D93H	3475	IA32_PASID		See Table 2-2.
1200H - 121FH	4608 - 4639	IA32_LBR_x_INFO		Last Branch Record Entry X Info Register (R/W) See Table 2-2.
1406H	5126	IA32_MCU_CONTROL		See Table 2-2.
14CEH	5326	IA32_LBR_CTL		Last Branch Record Enabling and Configuration Register (R/W) See Table 2-2.
14CFH	5327	IA32_LBR_DEPTH		Last Branch Record Maximum Stack Depth Register (R/W) See Table 2-2.
1500H - 151FH	5376 - 5407	IA32_LBR_x_FROM_IP		Last Branch Record Entry X Source IP Register (R/W) See Table 2-2.
1600H - 161FH	5632 - 5663	IA32_LBR_x_TO_IP		Last Branch Record Entry X Destination IP Register (R/W) See Table 2-2.

2.18 MSRS IN INTEL® XEON PHI™ PROCESSOR 3200/5200/7200 SERIES AND INTEL® XEON PHI™ PROCESSOR 7215/7285/7295 SERIES

Intel[®] Xeon Phi[™] processor 3200, 5200, 7200 series, with CPUID DisplayFamily_DisplayModel signature 06_57H, supports the MSR interfaces listed in Table 2-53. These processors are based on the Knights Landing microarchitecture. Intel[®] Xeon Phi[™] processor 7215, 7285, 7295 series, with CPUID DisplayFamily_DisplayModel signature 06_85H, supports the MSR interfaces listed in Table 2-53 and Table 2-54. These processors are based on the Knights Mill microarchitecture. Some MSRs are shared between a pair of processor cores, the scope is marked as module.

Table 2-53. Selected MSRs Supported by Intel® Xeon Phi™ Processors with DisplayFamily_DisplayModel Signatures 06_57H and 06_85H

	ister ress	Register Name / Bit Fields (Former MSR Name)	Scope	Bit Description
Hex	Dec			
OH	0	IA32_P5_MC_ADDR	Module	See Section 2.23, "MSRs in Pentium Processors."
1H	1	IA32_P5_MC_TYPE	Module	See Section 2.23, "MSRs in Pentium Processors."
6H	6	IA32_MONITOR_FILTER_SIZE	Thread	See Section 9.10.5, "Monitor/Mwait Address Range Determination." See Table 2-2.
10H	16	IA32_TIME_STAMP_COUNTER	Thread	See Section 18.17, "Time-Stamp Counter," and see Table 2-2.
17H	23	IA32_PLATFORM_ID	Package	Platform ID (R) See Table 2-2.
1BH	27	IA32_APIC_BASE	Thread	See Section 11.4.4, "Local APIC Status and Location," and Table 2-2.
34H	52	MSR_SMI_COUNT	Thread	SMI Counter (R/O)
		31:0		SMI Count (R/O)
		63:32		Reserved
ЗАН	58	IA32_FEATURE_CONTROL	Thread	Control Features in Intel 64Processor (R/W) See Table 2-2.
		0		Lock (R/WL)
		1		Reserved
		2		Enable VMX outside SMX operation (R/WL)
3BH	59	IA32_TSC_ADJUST	THREAD	Per-Logical-Processor TSC ADJUST (R/W) See Table 2-2.
4EH	78	IA32_PPIN_CTL (MSR_PPIN_CTL)	Package	Protected Processor Inventory Number Enable Control (R/W)
		0		LockOut (R/WO) See Table 2-2.
		1		Enable_PPIN (R/W)
				See Table 2-2.
		63:2		Reserved
4FH	79	IA32_PPIN (MSR_PPIN)	Package	Protected Processor Inventory Number (R/O)
		63:0		Protected Processor Inventory Number (R/O) See Table 2-2.

Table 2-53. Selected MSRs Supported by Intel® Xeon Phi™ Processors with DisplayFamily_DisplayModel Signatures 06_57H and 06_85H

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
79H	121	IA32_BIOS_UPDT_TRIG	Core	BIOS Update Trigger Register (W) See Table 2-2.
8BH	139	IA32_BIOS_SIGN_ID	THREAD	BIOS Update Signature ID (R/W) See Table 2-2.
C1H	193	IA32_PMC0	THREAD	Performance Counter Register See Table 2-2.
C2H	194	IA32_PMC1	THREAD	Performance Counter Register See Table 2-2.
CEH	206	MSR_PLATFORM_INFO	Package	Platform Information
				Contains power management and other model specific features enumeration. See http://biosbits.org.
		7:0		Reserved
		15:8	Package	Maximum Non-Turbo Ratio (R/O)
				This is the ratio of the frequency that invariant TSC runs at. Frequency = ratio * 100 MHz.
		27:16		Reserved
		28	Package	Programmable Ratio Limit for Turbo Mode (R/O) When set to 1, indicates that Programmable Ratio Limit for Turbo mode is enabled. When set to 0, indicates Programmable Ratio Limit for Turbo mode is disabled.
		29	Package	Programmable TDP Limit for Turbo Mode (R/O) When set to 1, indicates that TDP Limit for Turbo mode is programmable. When set to 0, indicates TDP Limit for Turbo mode is not programmable.
		39:30		Reserved
		47:40	Package	Maximum Efficiency Ratio (R/O)
				This is the minimum ratio (maximum efficiency) that the processor can operate, in units of 100MHz.
		63:48		Reserved
E2H	226	MSR_PKG_CST_CONFIG_CONTROL	Package	C-State Configuration Control (R/W)

Table 2-53. Selected MSRs Supported by Intel® Xeon Phi™ Processors with DisplayFamily_DisplayModel Signatures 06_57H and 06_85H

Regi Addı		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
		2:0		Package C-State Limit (R/W)
				Specifies the lowest C-state for the package. This feature does not limit the processor core C-state. The power-on default value from bit[2:0] of this register reports the deepest package C-state the processor is capable to support when manufactured. It is recommended that BIOS always read the power-on default value reported from this bit field to determine the supported deepest C-state on the processor and leave it as default without changing it. 000b - CO/C1 (No package C-state support) 001b - C2 010b - C6 (non retention)* 011b - C6 (Retention)* 100b - Reserved 111b - No package C-state limit. All C-States supported by the processor are available. Note: C6 retention mode provides more power saving than C6 non-retention mode. Limiting the package to C6 non retention mode does prevent the MSR_PKG_C6_RESIDENCY counter (MSR 3F9h) from
				being incremented.
		9:3		Reserved
		10		I/O MWAIT Redirection Enable (R/W) When set, will map IO_read instructions sent to IO registers at MSR_PMG_IO_CAPTURE_BASE[15:0] to MWAIT instructions.
		14:11		Reserved
		15		CFG Lock (R/O)
				When set, locks bits [15:0] of this register for further writes until the next reset occurs.
		25		Reserved
		26		C1 State Auto Demotion Enable (R/W)
				When set, the processor will conditionally demote C3/C6/C7 requests to C1 based on uncore auto-demote information.
		27		Reserved
		28		C1 State Auto Undemotion Enable (R/W) When set, enables Undemotion from Demoted C1.
		29		PKG C-State Auto Demotion Enable (R/W)
				When set, enables Package C state demotion.
		63:30		Reserved

Table 2-53. Selected MSRs Supported by Intel® Xeon Phi™ Processors with DisplayFamily_DisplayModel Signatures 06_57H and 06_85H

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
E4H	228	MSR_PMG_IO_CAPTURE_BASE	Tile	Power Management IO Capture Base (R/W)
		15:0		LVL_2 Base Address (R/W)
				Microcode will compare IO-read zone to this base address to determine if an MWAIT(C2/3/4) needs to be issued instead of the IO-read. Should be programmed to the chipset Plevel_2 IO address.
		22:16		C-State Range (R/W)
				The IO-port block size in which IO-redirection will be executed (0-127). Should be programmed based on the number of LVLx registers existing in the chipset.
		63:23		Reserved
E7H	231	IA32_MPERF	Thread	Maximum Performance Frequency Clock Count (R/W) See Table 2-2.
E8H	232	IA32_APERF	Thread	Actual Performance Frequency Clock Count (R/W) See Table 2-2.
FEH	254	IA32_MTRRCAP	Core	Memory Type Range Register (R)
				See Table 2-2.
13CH	316	MSR_FEATURE_CONFIG	Core	AES Configuration (RW-L)
				Privileged post-BIOS agent must provide a #GP handler to handle unsuccessful read of this MSR.
		1:0		AES Configuration (RW-L)
				Upon a successful read of this MSR, the configuration of AES instruction set availability is as follows:
				11b: AES instructions are not available until next RESET.
				Otherwise, AES instructions are available.
				Note, the AES instruction set is not available if read is unsuccessful. If the configuration is not 01b, AES instructions can be mis-configured if a privileged agent unintentionally writes 11b.
		63:2		Reserved
140H	320	MISC_FEATURE_ENABLES	Thread	MISC_FEATURE_ENABLES
		0		Reserved
		1		User Mode MONITOR and MWAIT (R/W)
				If set to 1, the MONITOR and MWAIT instructions do not cause invalid-opcode exceptions when executed with CPL > 0 or in virtual-8086 mode. If MWAIT is executed when CPL > 0 or in virtual-8086 mode, and if EAX indicates a C-state other than CO or C1, the instruction operates as if EAX indicated the C-state C1.
		63:2		Reserved
174H	372	IA32_SYSENTER_CS	Thread	See Table 2-2.
175H	373	IA32_SYSENTER_ESP	Thread	See Table 2-2.

Table 2-53. Selected MSRs Supported by Intel® Xeon Phi™ Processors with DisplayFamily_DisplayModel Signatures 06_57H and 06_85H

Regi Addı		Register Name / Bit Fields (Former MSR Name)	Scope	Bit Description
Hex	Dec			
176H	374	IA32_SYSENTER_EIP	Thread	See Table 2-2.
179H	377	IA32_MCG_CAP	Thread	See Table 2-2.
17AH	378	IA32_MCG_STATUS	Thread	See Table 2-2.
17DH	381	MSR_SMM_MCA_CAP	Thread	Enhanced SMM Capabilities (SMM-RO)
				Reports SMM capability Enhancement. Accessible only while in SMM.
		31:0		Bank Support (SMM-RO)
				One bit per MCA bank. If the bit is set, that bank supports Enhanced MCA (Default all 0; does not support EMCA).
		55:32		Reserved
		56		Targeted SMI (SMM-RO)
				Set if targeted SMI is supported.
		57		SMM_CPU_SVRSTR (SMM-R0)
				Set if SMM SRAM save/restore feature is supported.
		58		SMM_CODE_ACCESS_CHK (SMM-RO)
				Set if SMM code access check feature is supported.
		59		Long_Flow_Indication (SMM-RO) If set to 1, indicates that the SMM long flow indicator is supported and a host-space interface available to SMM handler.
		63:60		Reserved
186H	390	IA32_PERFEVTSEL0	Thread	Performance Monitoring Event Select Register (R/W) See Table 2-2.
		7:0		Event Select
		15:8		UMask
		16		USR
		17		OS OS
		18		Edge
		19		PC PC
		20		INT
		21		AnyThread
		22		EN
		23		INV
		31:24		CMASK
		63:32		Reserved
187H	391	IA32_PERFEVTSEL1	Thread	See Table 2-2.
198H	408	IA32_PERF_STATUS	Package	See Table 2-2.

Table 2-53. Selected MSRs Supported by Intel® Xeon Phi™ Processors with DisplayFamily_DisplayModel Signatures 06_57H and 06_85H

Regi Addı		Register Name / Bit Fields (Former MSR Name)	Scope	Bit Description
Hex	Dec			
199H	409	IA32_PERF_CTL	Thread	See Table 2-2.
19AH	410	IA32_CLOCK_MODULATION	Thread	Clock Modulation (R/W)
				See Table 2-2.
19BH	411	IA32_THERM_INTERRUPT	Module	Thermal Interrupt Control (R/W)
				See Table 2-2.
19CH	412	IA32_THERM_STATUS	Module	Thermal Monitor Status (R/W)
				See Table 2-2.
		0		Thermal Status (R/O)
		1		Thermal Status Log (R/WCO)
		2		PROTCHOT # or FORCEPR# Status (R/O)
		3		PROTCHOT # or FORCEPR# Log (R/WCO)
		4		Critical Temperature Status (R/O)
		5		Critical Temperature Status Log (R/WCO)
		6		Thermal Threshold #1 Status (R/O)
		7		Thermal Threshold #1 Log (R/WC0)
		8		Thermal Threshold #2 Status (R/O)
		9		Thermal Threshold #2 Log (R/WC0)
		10		Power Limitation Status (R/O)
		11		Power Limitation Log (RWCO)
		15:12		Reserved
		22:16		Digital Readout (R/O)
		26:23		Reserved
		30:27		Resolution in Degrees Celsius (R/O)
		31		Reading Valid (R/O)
		63:32		Reserved
1A0H	416	IA32_MISC_ENABLE	Thread	Enable Misc. Processor Features (R/W)
				Allows a variety of processor functions to be enabled and disabled.
		0		Fast-Strings Enable
		2:1		Reserved
		3		Automatic Thermal Control Circuit Enable (R/W)
		6:4		Reserved
		7		Performance Monitoring Available (R)
		10:8		Reserved
		11		Branch Trace Storage Unavailable (R/O)
		12		Processor Event Based Sampling Unavailable (R/O)
		15:13		Reserved

Table 2-53. Selected MSRs Supported by Intel® Xeon Phi™ Processors with DisplayFamily_DisplayModel Signatures 06_57H and 06_85H

	ister Iress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
		16		Enhanced Intel SpeedStep Technology Enable (R/W)
		18		ENABLE MONITOR FSM (R/W)
		21:19		Reserved
		22		Limit CPUID Maxval (R/W)
		23		xTPR Message Disable (R/W)
		33:24		Reserved
		34		XD Bit Disable (R/W)
		37:35		Reserved
		38		Turbo Mode Disable (R/W)
		63:39		Reserved
1A2H	418	MSR_TEMPERATURE_TARGET	Package	Temperature Target
		15:0		Reserved
		23:16		Temperature Target (R)
		29:24		Target Offset (R/W)
		63:30		Reserved
1A4H	420	MSR_MISC_FEATURE_CONTROL		Miscellaneous Feature Control (R/W)
		0	Core	DCU Hardware Prefetcher Disable (R/W)
				If 1, disables the L1 data cache prefetcher.
		1	Соге	L2 Hardware Prefetcher Disable (R/W)
				If 1, disables the L2 hardware prefetcher.
		63:2		Reserved
1A6H	422	MSR_OFFCORE_RSP_0	Shared	Offcore Response Event Select Register (R/W)
1A7H	423	MSR_OFFCORE_RSP_1	Shared	Offcore Response Event Select Register (R/W)
1ADH	429	MSR_TURBO_RATIO_LIMIT	Package	Maximum Ratio Limit of Turbo Mode for Groups of Cores (R/W)
		0		Reserved
		7:1	Package	Maximum Number of Cores in Group 0
				Number active processor cores which operates under the maximum ratio limit for group 0.
		15:8	Package	Maximum Ratio Limit for Group 0
				Maximum turbo ratio limit when the number of active cores are not more than the group 0 maximum core count.
		20:16	Package	Number of Incremental Cores Added to Group 1
				Group 1, which includes the specified number of additional cores plus the cores in group 0, operates under the group 1 turbo max ratio limit = "group 0 Max ratio limit" - "group ratio delta for group 1".

Table 2-53. Selected MSRs Supported by Intel® Xeon Phi™ Processors with DisplayFamily_DisplayModel Signatures 06_57H and 06_85H

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
		23:21	Package	Group Ratio Delta for Group 1 An unsigned integer specifying the ratio decrement relative to the Max ratio limit to Group 0.
		28:24	Package	Number of Incremental Cores Added to Group 2 Group 2, which includes the specified number of additional cores plus all the cores in group 1, operates under the group 2 turbo max ratio limit = "group 1 Max"
		31:29	Package	ratio limit" - "group ratio delta for group 2". Group Ratio Delta for Group 2
				An unsigned integer specifying the ratio decrement relative to the Max ratio limit for Group 1.
		36:32	Package	Number of Incremental Cores Added to Group 3 Group 3, which includes the specified number of additional cores plus all the cores in group 2, operates under the group 3 turbo max ratio limit = "group 2 Max ratio limit" - "group ratio delta for group 3".
		39:37	Package	Group Ratio Delta for Group 3 An unsigned integer specifying the ratio decrement relative to the Max ratio limit for Group 2.
		44:40	Package	Number of Incremental Cores Added to Group 4 Group 4, which includes the specified number of additional cores plus all the cores in group 3, operates under the group 4 turbo max ratio limit = "group 3 Max ratio limit" - "group ratio delta for group 4".
		47:45	Package	Group Ratio Delta for Group 4 An unsigned integer specifying the ratio decrement relative to the Max ratio limit for Group 3.
		52:48	Package	Number of Incremental Cores Added to Group 5 Group 5, which includes the specified number of additional cores plus all the cores in group 4, operates under the group 5 turbo max ratio limit = "group 4 Max ratio limit" - "group ratio delta for group 5".
		55:53	Package	Group Ratio Delta for Group 5 An unsigned integer specifying the ratio decrement relative to the Max ratio limit for Group 4.
		60:56	Package	Number of Incremental Cores Added to Group 6 Group 6, which includes the specified number of additional cores plus all the cores in group 5, operates under the group 6 turbo max ratio limit = "group 5 Max ratio limit" - "group ratio delta for group 6".
		63:61	Package	Group Ratio Delta for Group 6 An unsigned integer specifying the ratio decrement relative to the Max ratio limit for Group 5.
1B0H	432	IA32_ENERGY_PERF_BIAS	Thread	See Table 2-2.

Table 2-53. Selected MSRs Supported by Intel® Xeon Phi™ Processors with DisplayFamily_DisplayModel Signatures 06_57H and 06_85H

Regi: Addr		Register Name / Bit Fields (Former MSR Name) IA32_PACKAGE_THERM_STATUS	Scope	Bit Description
Hex	Dec			
1B1H	433	IA32_PACKAGE_THERM_STATUS	Package	See Table 2-2.
1B2H	434	IA32_PACKAGE_THERM_INTERRUPT	Package	See Table 2-2.
1C8H	456	MSR_LBR_SELECT	Thread	Last Branch Record Filtering Select Register (R/W)
				See Section 18.9.2, "Filtering of Last Branch Records."
		0		CPL_EQ_0
		1		CPL_NEQ_0
		2		JCC
		3		NEAR_REL_CALL
		4		NEAR_IND_CALL
		5		NEAR_RET
		6		NEAR_IND_JMP
		7		NEAR_REL_JMP
		8		FAR_BRANCH
		63:9		Reserved
1C9H	457	MSR_LASTBRANCH_TOS	Thread	Last Branch Record Stack TOS (R/W)
				Contains an index (bits 0-2) that points to the MSR
				containing the most recent branch record. See MSR_LASTBRANCH_0_FROM_IP.
1D9H	473	IA32_DEBUGCTL	Thread	Debug Control (R/W)
1090	4/3	0	Tilleau	LBR
		U		Setting this bit to 1 enables the processor to record a
				running trace of the most recent branches taken by the processor in the LBR stack.
		1		BTF
				Setting this bit to 1 enables the processor to treat EFLAGS.TF as single-step on branches instead of single-step on instructions.
		5:2		Reserved
		6		TR
				Setting this bit to 1 enables branch trace messages to be sent.
		7		BTS
				Setting this bit enables branch trace messages (BTMs) to be logged in a BTS buffer.
		8		BTINT
				When clear, BTMs are logged in a BTS buffer in circular fashion. When this bit is set, an interrupt is generated by the BTS facility when the BTS buffer is full.
		9		BTS_OFF_OS
				When set, BTS or BTM is skipped if CPL = 0.

Table 2-53. Selected MSRs Supported by Intel® Xeon Phi™ Processors with DisplayFamily_DisplayModel Signatures 06_57H and 06_85H

Hex Dec (Former MSR Name) BTS_OFF_USR When set, BTS or BTM is skipped if CPL > 0. 11 REEZE_LBRS_ON_PMI When set, BTS or BTM is skipped if CPL > 0. 12 FREEZE_LBRS_ON_PMI When set, be LBR stack is frozen on a PMI request. 13 RESERVED 14 FREEZE_WHITLE_SMM When set, seach ENABLE bit of the global counter control MSR are frozen (address 38FH) on a PMI request. 1DDH 477 MSR_LER_FROM_LIP Thread 1FEH 498 132_SMRR_PHYSBASE Core 152 498. IA32_SMRR_PHYSBASE 164 479 175 Thread 186 2-2. 200H 512 183_SMRR_PHYSBASE Core 200H 512 184_ABZ_MTRR_PHYSBASEO Core 201H 513 182_MTRR_PHYSBASEO Core 202H 514 183_MTRR_PHYSBASEO Core 203H 515 184_ABZ_MTRR_PHYSBASE1 Core 204H 516 184_ABZ_MTRR_PHYSBASE2 Core </th <th>_</th> <th>ister ress</th> <th>Register Name / Bit Fields</th> <th>Scope</th> <th>Bit Description</th>	_	ister ress	Register Name / Bit Fields	Scope	Bit Description
LAND AND AND AND AND AND AND AND AND AND	Hex	Dec	(Former MSR Name)		
11			10		BTS_OFF_USR
Lange of the control of the					When set, BTS or BTM is skipped if CPL > 0.
12			11		
When set, each ENABLE bit of the global counter control MSR are frozen (address 3BFH) on a PMI request.					·
MSR are frozen (address 3BFH) on a PMI request.			12		
Table					
When set, freezes perfmon and trace messages while in SMM.			13		Reserved
SMM. Reserved Re			14		FREEZE_WHILE_SMM
IDDH 477 MSR_LER_FROM_LIP Thread Last Exception Record from Linear IP (R) 1DEH 478 MSR_LER_TO_LIP Thread Last Exception Record to Linear IP (R) 1F2H 498 IA32_SMRR_PHYSBASE Core See Table 2-2. 1F3H 499 IA32_SMRR_PHYSMASK Core See Table 2-2. 200H 512 IA32_MTRR_PHYSBASE0 Core See Table 2-2. 201H 513 IA32_MTRR_PHYSMASK0 Core See Table 2-2. 202H 514 IA32_MTRR_PHYSMASK1 Core See Table 2-2. 203H 515 IA32_MTRR_PHYSMASK1 Core See Table 2-2. 204H 516 IA32_MTRR_PHYSMASK2 Core See Table 2-2. 205H 517 IA32_MTRR_PHYSMASK2 Core See Table 2-2. 206H 518 IA32_MTRR_PHYSMASK3 Core See Table 2-2. 209H 520 IA32_MTRR_PHYSMASK4 Core See Table 2-2. 209H 521 IA32_MTRR_PHYSMASK5 Core See Table 2-2.					When set, freezes perfmon and trace messages while in SMM.
1DEH 478 MSR_LER_TO_LIP Thread Last Exception Record to Linear IP (R) 1F2H 498 IA32_SMRR_PHYSBASE Core See Table 2-2. 1F3H 499 IA32_SMRR_PHYSMASK Core See Table 2-2. 200H 512 IA32_MTRR_PHYSBASEO Core See Table 2-2. 201H 513 IA32_MTRR_PHYSMASKO Core See Table 2-2. 202H 514 IA32_MTRR_PHYSBASE1 Core See Table 2-2. 203H 515 IA32_MTRR_PHYSMASK1 Core See Table 2-2. 204H 516 IA32_MTRR_PHYSBASE2 Core See Table 2-2. 205H 517 IA32_MTRR_PHYSBASE3 Core See Table 2-2. 206H 518 IA32_MTRR_PHYSBASE3 Core See Table 2-2. 207H 519 IA32_MTRR_PHYSBASE4 Core See Table 2-2. 208H 520 IA32_MTRR_PHYSBASE5 Core See Table 2-2. 209H 521 IA32_MTRR_PHYSBASE5 Core See Table 2-2. 200H			31:15		Reserved
1F2H 498 IA32_SMRR_PHYSBASE Core See Table 2-2. 1F3H 499 IA32_SMRR_PHYSMASK Core See Table 2-2. 200H 512 IA32_MTRR_PHYSBASEO Core See Table 2-2. 201H 513 IA32_MTRR_PHYSMASKO Core See Table 2-2. 202H 514 IA32_MTRR_PHYSBASE1 Core See Table 2-2. 203H 515 IA32_MTRR_PHYSMASK1 Core See Table 2-2. 204H 516 IA32_MTRR_PHYSBASE2 Core See Table 2-2. 205H 517 IA32_MTRR_PHYSMASK2 Core See Table 2-2. 206H 518 IA32_MTRR_PHYSBASE3 Core See Table 2-2. 207H 519 IA32_MTRR_PHYSBASE3 Core See Table 2-2. 208H 520 IA32_MTRR_PHYSBASE4 Core See Table 2-2. 209H 521 IA32_MTRR_PHYSBASE5 Core See Table 2-2. 208H 523 IA32_MTRR_PHYSBASE6 Core See Table 2-2. 200H 525	1DDH	477	MSR_LER_FROM_LIP	Thread	Last Exception Record from Linear IP (R)
1F3H 499 IA32_SMRR_PHYSMASK Core See Table 2-2. 200H 512 IA32_MTRR_PHYSMASKO Core See Table 2-2. 201H 513 IA32_MTRR_PHYSMASKO Core See Table 2-2. 202H 514 IA32_MTRR_PHYSMASK1 Core See Table 2-2. 203H 515 IA32_MTRR_PHYSMASK1 Core See Table 2-2. 204H 516 IA32_MTRR_PHYSMASK2 Core See Table 2-2. 205H 517 IA32_MTRR_PHYSMASK2 Core See Table 2-2. 206H 518 IA32_MTRR_PHYSMASK3 Core See Table 2-2. 207H 519 IA32_MTRR_PHYSMASK3 Core See Table 2-2. 208H 520 IA32_MTRR_PHYSMASK4 Core See Table 2-2. 209H 521 IA32_MTRR_PHYSMASK5 Core See Table 2-2. 208H 523 IA32_MTRR_PHYSMASK5 Core See Table 2-2. 20BH 523 IA32_MTRR_PHYSMASK5 Core See Table 2-2. 20CH 526	1DEH	478	MSR_LER_TO_LIP	Thread	Last Exception Record to Linear IP (R)
200H 512 IA32_MTRR_PHYSBASE0 Core See Table 2-2. 201H 513 IA32_MTRR_PHYSMASK0 Core See Table 2-2. 202H 514 IA32_MTRR_PHYSBASE1 Core See Table 2-2. 203H 515 IA32_MTRR_PHYSMASK1 Core See Table 2-2. 204H 516 IA32_MTRR_PHYSMASE2 Core See Table 2-2. 205H 517 IA32_MTRR_PHYSMASK2 Core See Table 2-2. 206H 518 IA32_MTRR_PHYSMASK3 Core See Table 2-2. 207H 519 IA32_MTRR_PHYSMASK3 Core See Table 2-2. 208H 520 IA32_MTRR_PHYSMASK3 Core See Table 2-2. 209H 521 IA32_MTRR_PHYSMASK4 Core See Table 2-2. 208H 522 IA32_MTRR_PHYSMASK5 Core See Table 2-2. 20BH 523 IA32_MTRR_PHYSMASK6 Core See Table 2-2. 20DH 525 IA32_MTRR_PHYSMASK6 Core See Table 2-2. 20FH 527	1F2H	498	IA32_SMRR_PHYSBASE	Соге	See Table 2-2.
201H 513 IA32_MTRR_PHYSMASKO Core See Table 2-2. 202H 514 IA32_MTRR_PHYSBASE1 Core See Table 2-2. 203H 515 IA32_MTRR_PHYSMASK1 Core See Table 2-2. 204H 516 IA32_MTRR_PHYSMASK2 Core See Table 2-2. 205H 517 IA32_MTRR_PHYSMASK2 Core See Table 2-2. 206H 518 IA32_MTRR_PHYSMASK3 Core See Table 2-2. 207H 519 IA32_MTRR_PHYSMASK3 Core See Table 2-2. 208H 520 IA32_MTRR_PHYSMASK4 Core See Table 2-2. 209H 521 IA32_MTRR_PHYSMASK4 Core See Table 2-2. 208H 522 IA32_MTRR_PHYSMASK5 Core See Table 2-2. 208H 523 IA32_MTRR_PHYSMASK5 Core See Table 2-2. 20CH 524 IA32_MTRR_PHYSMASK6 Core See Table 2-2. 20CH 525 IA32_MTRR_PHYSMASK7 Core See Table 2-2. 20FH 527	1F3H	499	IA32_SMRR_PHYSMASK	Core	See Table 2-2.
202H 514 IA32_MTRR_PHYSBASE1 Core See Table 2-2. 203H 515 IA32_MTRR_PHYSMASK1 Core See Table 2-2. 204H 516 IA32_MTRR_PHYSBASE2 Core See Table 2-2. 205H 517 IA32_MTRR_PHYSMASK2 Core See Table 2-2. 206H 518 IA32_MTRR_PHYSBASE3 Core See Table 2-2. 207H 519 IA32_MTRR_PHYSMASK3 Core See Table 2-2. 208H 520 IA32_MTRR_PHYSBASE4 Core See Table 2-2. 209H 521 IA32_MTRR_PHYSMASK4 Core See Table 2-2. 208H 522 IA32_MTRR_PHYSBASE5 Core See Table 2-2. 208H 523 IA32_MTRR_PHYSBASE6 Core See Table 2-2. 20CH 524 IA32_MTRR_PHYSBASE6 Core See Table 2-2. 20CH 525 IA32_MTRR_PHYSBASE7 Core See Table 2-2. 20FH 527 IA32_MTRR_PHYSBASE7 Core See Table 2-2. 20FH 526	200H	512	IA32_MTRR_PHYSBASE0	Соге	See Table 2-2.
203H 515 IA32_MTRR_PHYSMASK1 Core See Table 2-2. 204H 516 IA32_MTRR_PHYSBASE2 Core See Table 2-2. 205H 517 IA32_MTRR_PHYSMASK2 Core See Table 2-2. 206H 518 IA32_MTRR_PHYSBASE3 Core See Table 2-2. 207H 519 IA32_MTRR_PHYSMASK3 Core See Table 2-2. 208H 520 IA32_MTRR_PHYSBASE4 Core See Table 2-2. 209H 521 IA32_MTRR_PHYSMASK4 Core See Table 2-2. 20AH 522 IA32_MTRR_PHYSBASE5 Core See Table 2-2. 20BH 523 IA32_MTRR_PHYSBASE6 Core See Table 2-2. 20CH 524 IA32_MTRR_PHYSBASE6 Core See Table 2-2. 20CH 525 IA32_MTRR_PHYSBASE7 Core See Table 2-2. 20FH 526 IA32_MTRR_PHYSBASE7 Core See Table 2-2. 20FH 527 IA32_MTRR_FIX16K_00000 Core See Table 2-2. 259H 600	201H	513	IA32_MTRR_PHYSMASK0	Соге	See Table 2-2.
204H 516 IA32_MTRR_PHYSBASE2 Core See Table 2-2. 205H 517 IA32_MTRR_PHYSMASK2 Core See Table 2-2. 206H 518 IA32_MTRR_PHYSBASE3 Core See Table 2-2. 207H 519 IA32_MTRR_PHYSMASK3 Core See Table 2-2. 208H 520 IA32_MTRR_PHYSBASE4 Core See Table 2-2. 209H 521 IA32_MTRR_PHYSMASK4 Core See Table 2-2. 20AH 522 IA32_MTRR_PHYSMASK5 Core See Table 2-2. 20BH 523 IA32_MTRR_PHYSMASK5 Core See Table 2-2. 20CH 524 IA32_MTRR_PHYSMASK6 Core See Table 2-2. 20DH 525 IA32_MTRR_PHYSMASK6 Core See Table 2-2. 20FH 526 IA32_MTRR_PHYSMASK7 Core See Table 2-2. 20FH 527 IA32_MTRR_PHYSMASK7 Core See Table 2-2. 259H 592 IA32_MTRR_FIX64K_00000 Core See Table 2-2. 258H 600	202H	514	IA32_MTRR_PHYSBASE1	Соге	See Table 2-2.
205H 517 IA32_MTRR_PHYSMASK2 Core See Table 2-2. 206H 518 IA32_MTRR_PHYSBASE3 Core See Table 2-2. 207H 519 IA32_MTRR_PHYSMASK3 Core See Table 2-2. 208H 520 IA32_MTRR_PHYSBASE4 Core See Table 2-2. 209H 521 IA32_MTRR_PHYSMASK4 Core See Table 2-2. 20AH 522 IA32_MTRR_PHYSBASE5 Core See Table 2-2. 20BH 523 IA32_MTRR_PHYSMASK5 Core See Table 2-2. 20CH 524 IA32_MTRR_PHYSBASE6 Core See Table 2-2. 20DH 525 IA32_MTRR_PHYSBASE7 Core See Table 2-2. 20EH 526 IA32_MTRR_PHYSBASE7 Core See Table 2-2. 25DH 592 IA32_MTRR_FIX64K_00000 Core See Table 2-2. 25BH 600 IA32_MTRR_FIX16K_80000 Core See Table 2-2. 259H 601 IA32_MTRR_FIX16K_0000 Core See Table 2-2. 25BH 616<	203H	515	IA32_MTRR_PHYSMASK1	Соге	See Table 2-2.
206H 518 IA32_MTRR_PHYSBASE3 Core See Table 2-2. 207H 519 IA32_MTRR_PHYSMASK3 Core See Table 2-2. 208H 520 IA32_MTRR_PHYSBASE4 Core See Table 2-2. 209H 521 IA32_MTRR_PHYSMASK4 Core See Table 2-2. 20AH 522 IA32_MTRR_PHYSBASE5 Core See Table 2-2. 20BH 523 IA32_MTRR_PHYSMASK5 Core See Table 2-2. 20CH 524 IA32_MTRR_PHYSBASE6 Core See Table 2-2. 20DH 525 IA32_MTRR_PHYSBASE7 Core See Table 2-2. 20EH 526 IA32_MTRR_PHYSMASK7 Core See Table 2-2. 25DH 592 IA32_MTRR_FIX64K_00000 Core See Table 2-2. 259H 600 IA32_MTRR_FIX16K_80000 Core See Table 2-2. 259H 601 IA32_MTRR_FIX16K_A0000 Core See Table 2-2. 268H 616 IA32_MTRR_FIX4K_C0000 Core See Table 2-2.	204H	516	IA32_MTRR_PHYSBASE2	Соге	See Table 2-2.
207H 519 IA32_MTRR_PHYSMASK3 Core See Table 2-2. 208H 520 IA32_MTRR_PHYSBASE4 Core See Table 2-2. 209H 521 IA32_MTRR_PHYSMASK4 Core See Table 2-2. 20AH 522 IA32_MTRR_PHYSBASE5 Core See Table 2-2. 20BH 523 IA32_MTRR_PHYSMASK5 Core See Table 2-2. 20CH 524 IA32_MTRR_PHYSBASE6 Core See Table 2-2. 20DH 525 IA32_MTRR_PHYSBASE7 Core See Table 2-2. 20EH 526 IA32_MTRR_PHYSMASK7 Core See Table 2-2. 20FH 527 IA32_MTRR_PHYSMASK7 Core See Table 2-2. 250H 592 IA32_MTRR_FIX64K_00000 Core See Table 2-2. 258H 600 IA32_MTRR_FIX16K_80000 Core See Table 2-2. 259H 601 IA32_MTRR_FIX16K_00000 Core See Table 2-2. 268H 616 IA32_MTRR_FIX4K_C0000 Core See Table 2-2.	205H	517	IA32_MTRR_PHYSMASK2	Соге	See Table 2-2.
208H 520 IA32_MTRR_PHYSBASE4 Core See Table 2-2. 209H 521 IA32_MTRR_PHYSMASK4 Core See Table 2-2. 20AH 522 IA32_MTRR_PHYSBASE5 Core See Table 2-2. 20BH 523 IA32_MTRR_PHYSMASK5 Core See Table 2-2. 20CH 524 IA32_MTRR_PHYSBASE6 Core See Table 2-2. 20DH 525 IA32_MTRR_PHYSMASK6 Core See Table 2-2. 20EH 526 IA32_MTRR_PHYSBASE7 Core See Table 2-2. 20FH 527 IA32_MTRR_PHYSMASK7 Core See Table 2-2. 250H 592 IA32_MTRR_FIX64K_00000 Core See Table 2-2. 258H 600 IA32_MTRR_FIX16K_80000 Core See Table 2-2. 259H 601 IA32_MTRR_FIX16K_A0000 Core See Table 2-2. 268H 616 IA32_MTRR_FIX4K_C0000 Core See Table 2-2.	206H	518	IA32_MTRR_PHYSBASE3	Соге	See Table 2-2.
209H 521 IA32_MTRR_PHYSMASK4 Core See Table 2-2. 20AH 522 IA32_MTRR_PHYSBASE5 Core See Table 2-2. 20BH 523 IA32_MTRR_PHYSMASK5 Core See Table 2-2. 20CH 524 IA32_MTRR_PHYSBASE6 Core See Table 2-2. 20DH 525 IA32_MTRR_PHYSMASK6 Core See Table 2-2. 20EH 526 IA32_MTRR_PHYSMASK7 Core See Table 2-2. 20FH 527 IA32_MTRR_PHYSMASK7 Core See Table 2-2. 250H 592 IA32_MTRR_FIX64K_00000 Core See Table 2-2. 258H 600 IA32_MTRR_FIX16K_80000 Core See Table 2-2. 259H 601 IA32_MTRR_FIX16K_A0000 Core See Table 2-2. 268H 616 IA32_MTRR_FIX4K_C0000 Core See Table 2-2.	207H	519	IA32_MTRR_PHYSMASK3	Соге	See Table 2-2.
20AH 522 IA32_MTRR_PHYSBASE5 Core See Table 2-2. 20BH 523 IA32_MTRR_PHYSMASK5 Core See Table 2-2. 20CH 524 IA32_MTRR_PHYSBASE6 Core See Table 2-2. 20DH 525 IA32_MTRR_PHYSMASK6 Core See Table 2-2. 20EH 526 IA32_MTRR_PHYSMASK7 Core See Table 2-2. 20FH 527 IA32_MTRR_PHYSMASK7 Core See Table 2-2. 250H 592 IA32_MTRR_FIX64K_00000 Core See Table 2-2. 258H 600 IA32_MTRR_FIX16K_80000 Core See Table 2-2. 259H 601 IA32_MTRR_FIX16K_A0000 Core See Table 2-2. 268H 616 IA32_MTRR_FIX4K_C0000 Core See Table 2-2.	208H	520	IA32_MTRR_PHYSBASE4	Соге	See Table 2-2.
20BH 523 IA32_MTRR_PHYSMASK5 Core See Table 2-2. 20CH 524 IA32_MTRR_PHYSBASE6 Core See Table 2-2. 20DH 525 IA32_MTRR_PHYSMASK6 Core See Table 2-2. 20EH 526 IA32_MTRR_PHYSBASE7 Core See Table 2-2. 20FH 527 IA32_MTRR_PHYSMASK7 Core See Table 2-2. 250H 592 IA32_MTRR_FIX64K_00000 Core See Table 2-2. 258H 600 IA32_MTRR_FIX16K_80000 Core See Table 2-2. 259H 601 IA32_MTRR_FIX16K_A0000 Core See Table 2-2. 268H 616 IA32_MTRR_FIX4K_C0000 Core See Table 2-2.	209H	521	IA32_MTRR_PHYSMASK4	Соге	See Table 2-2.
20CH 524 IA32_MTRR_PHYSBASE6 Core See Table 2-2. 20DH 525 IA32_MTRR_PHYSMASK6 Core See Table 2-2. 20EH 526 IA32_MTRR_PHYSBASE7 Core See Table 2-2. 20FH 527 IA32_MTRR_PHYSMASK7 Core See Table 2-2. 250H 592 IA32_MTRR_FIX64K_00000 Core See Table 2-2. 258H 600 IA32_MTRR_FIX16K_80000 Core See Table 2-2. 259H 601 IA32_MTRR_FIX16K_A0000 Core See Table 2-2. 268H 616 IA32_MTRR_FIX4K_C0000 Core See Table 2-2.	20AH	522	IA32_MTRR_PHYSBASE5	Соге	See Table 2-2.
20DH 525 IA32_MTRR_PHYSMASK6 Core See Table 2-2. 20EH 526 IA32_MTRR_PHYSBASE7 Core See Table 2-2. 20FH 527 IA32_MTRR_PHYSMASK7 Core See Table 2-2. 250H 592 IA32_MTRR_FIX64K_00000 Core See Table 2-2. 258H 600 IA32_MTRR_FIX16K_80000 Core See Table 2-2. 259H 601 IA32_MTRR_FIX16K_A0000 Core See Table 2-2. 268H 616 IA32_MTRR_FIX4K_C0000 Core See Table 2-2.	20BH	523	IA32_MTRR_PHYSMASK5	Соге	See Table 2-2.
20EH 526 IA32_MTRR_PHYSBASE7 Core See Table 2-2. 20FH 527 IA32_MTRR_PHYSMASK7 Core See Table 2-2. 250H 592 IA32_MTRR_FIX64K_00000 Core See Table 2-2. 258H 600 IA32_MTRR_FIX16K_80000 Core See Table 2-2. 259H 601 IA32_MTRR_FIX16K_A0000 Core See Table 2-2. 268H 616 IA32_MTRR_FIX4K_C0000 Core See Table 2-2.	20CH	524	IA32_MTRR_PHYSBASE6	Соге	See Table 2-2.
20FH 527 IA32_MTRR_PHYSMASK7 Core See Table 2-2. 250H 592 IA32_MTRR_FIX64K_00000 Core See Table 2-2. 258H 600 IA32_MTRR_FIX16K_80000 Core See Table 2-2. 259H 601 IA32_MTRR_FIX16K_A0000 Core See Table 2-2. 268H 616 IA32_MTRR_FIX4K_C0000 Core See Table 2-2.	20DH	525	IA32_MTRR_PHYSMASK6	Соге	See Table 2-2.
250H 592 IA32_MTRR_FIX64K_00000 Core See Table 2-2. 258H 600 IA32_MTRR_FIX16K_80000 Core See Table 2-2. 259H 601 IA32_MTRR_FIX16K_A0000 Core See Table 2-2. 268H 616 IA32_MTRR_FIX4K_C0000 Core See Table 2-2.	20EH	526	IA32_MTRR_PHYSBASE7	Core	See Table 2-2.
258H 600 IA32_MTRR_FIX16K_80000 Core See Table 2-2. 259H 601 IA32_MTRR_FIX16K_A0000 Core See Table 2-2. 268H 616 IA32_MTRR_FIX4K_C0000 Core See Table 2-2.	20FH	527	IA32_MTRR_PHYSMASK7	Core	See Table 2-2.
259H 601 IA32_MTRR_FIX16K_A0000 Core See Table 2-2. 268H 616 IA32_MTRR_FIX4K_C0000 Core See Table 2-2.	250H	592	IA32_MTRR_FIX64K_00000	Core	See Table 2-2.
268H 616 IA32_MTRR_FIX4K_C0000 Core See Table 2-2.	258H	600	IA32_MTRR_FIX16K_80000	Core	See Table 2-2.
268H 616 IA32_MTRR_FIX4K_C0000 Core See Table 2-2.	259H	601	IA32_MTRR_FIX16K_A0000	Core	See Table 2-2.
269H 617 IA32_MTRR_FIX4K_C8000 Core See Table 2-2.	268H	616	IA32_MTRR_FIX4K_C0000	Core	See Table 2-2.
	269H	617	IA32_MTRR_FIX4K_C8000	Core	See Table 2-2.

Table 2-53. Selected MSRs Supported by Intel® Xeon Phi™ Processors with DisplayFamily_DisplayModel Signatures 06_57H and 06_85H

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
26AH	618	IA32_MTRR_FIX4K_D0000	Core	See Table 2-2.
26BH	619	IA32_MTRR_FIX4K_D8000	Core	See Table 2-2.
26CH	620	IA32_MTRR_FIX4K_E0000	Core	See Table 2-2.
26DH	621	IA32_MTRR_FIX4K_E8000	Core	See Table 2-2.
26EH	622	IA32_MTRR_FIX4K_F0000	Core	See Table 2-2.
26FH	623	IA32_MTRR_FIX4K_F8000	Core	See Table 2-2.
277H	631	IA32_PAT	Core	See Table 2-2.
2FFH	767	IA32_MTRR_DEF_TYPE	Core	Default Memory Types (R/W) See Table 2-2.
309H	777	IA32_FIXED_CTR0	Thread	Fixed-Function Performance Counter Register 0 (R/W) See Table 2-2.
30AH	778	IA32_FIXED_CTR1	Thread	Fixed-Function Performance Counter Register 1 (R/W) See Table 2-2.
30BH	779	IA32_FIXED_CTR2	Thread	Fixed-Function Performance Counter Register 2 (R/W) See Table 2-2.
345H	837	IA32_PERF_CAPABILITIES	Package	See Table 2-2. See Section 18.4.1, "IA32_DEBUGCTL MSR."
38DH	909	IA32_FIXED_CTR_CTRL	Thread	Fixed-Function-Counter Control Register (R/W) See Table 2-2.
38EH	910	IA32_PERF_GLOBAL_STATUS	Thread	See Table 2-2.
38FH	911	IA32_PERF_GLOBAL_CTRL	Thread	See Table 2-2.
390H	912	IA32_PERF_GLOBAL_OVF_CTRL	Thread	See Table 2-2.
3F1H	1009	IA32_PEBS_ENABLE (MSR_PEBS_ENABLE)	Thread	See Table 2-2.
3F8H	1016	MSR_PKG_C3_RESIDENCY	Package	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-states.
		63:0		Package C3 Residency Counter (R/O)
3F9H	1017	MSR_PKG_C6_RESIDENCY	Package	
		63:0		Package C6 Residency Counter (R/O)
3FAH	1018	MSR_PKG_C7_RESIDENCY	Package	
		63:0		Package C7 Residency Counter (R/O)
3FCH	1020	MSR_MCO_RESIDENCY	Module	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-states.
		63:0		Module CO Residency Counter (R/O)
3FDH	1021	MSR_MC6_RESIDENCY	Module	
		63:0		Module C6 Residency Counter (R/O)

Table 2-53. Selected MSRs Supported by Intel® Xeon Phi™ Processors with DisplayFamily_DisplayModel Signatures 06_57H and 06_85H

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
3FFH	1023	MSR_CORE_C6_RESIDENCY	Core	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-states.
		63:0		CORE C6 Residency Counter (R/O)
400H	1024	IA32_MC0_CTL	Соге	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
401H	1025	IA32_MC0_STATUS	Соге	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
402H	1026	IA32_MCO_ADDR	Соге	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
404H	1028	IA32_MC1_CTL	Core	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
405H	1029	IA32_MC1_STATUS	Соге	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
408H	1032	IA32_MC2_CTL	Core	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
409H	1033	IA32_MC2_STATUS	Соге	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
40AH	1034	IA32_MC2_ADDR	Соге	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
40CH	1036	IA32_MC3_CTL	Соге	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
40DH	1037	IA32_MC3_STATUS	Соге	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
40EH	1038	IA32_MC3_ADDR	Соге	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
410H	1040	IA32_MC4_CTL	Соге	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
411H	1041	IA32_MC4_STATUS	Соге	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
412H	1042	IA32_MC4_ADDR	Core	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
				The MSR_MC4_ADDR register is either not implemented or contains no address if the ADDRV flag in the MSR_MC4_STATUS register is clear.
				When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
414H	1044	IA32_MC5_CTL	Package	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
415H	1045	IA32_MC5_STATUS	Package	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
416H	1046	IA32_MC5_ADDR	Package	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
4C1H	1217	IA32_A_PMC0	Thread	See Table 2-2.
4C2H	1218	IA32_A_PMC1	Thread	See Table 2-2.
600H	1536	IA32_DS_AREA	Thread	DS Save Area (R/W)
				See Table 2-2.
606H	1542	MSR_RAPL_POWER_UNIT	Package	Unit Multipliers Used in RAPL Interfaces (R/O)
		3:0	Package	Power Units
				See Section 15.10.1, "RAPL Interfaces."
		7:4	Package	Reserved

Table 2-53. Selected MSRs Supported by Intel® Xeon Phi™ Processors with DisplayFamily_DisplayModel Signatures 06_57H and 06_85H

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
		12:8	Package	Energy Status Units Energy related information (in Joules) is based on the multiplier, 1/2^ESU; where ESU is an unsigned integer represented by bits 12:8. Default value is 0EH (or 61 micro-joules).
		15:13	Package	Reserved
		19:16	Package	Time Units See Section 15.10.1, "RAPL Interfaces."
		63:20		Reserved
60DH	1549	MSR_PKG_C2_RESIDENCY	Package	Note: C-state values are processor specific C-state code names, unrelated to MWAIT extension C-state parameters or ACPI C-states.
		63:0		Package C2 Residency Counter (R/O)
610H	1552	MSR_PKG_POWER_LIMIT	Package	PKG RAPL Power Limit Control (R/W)
				See Section 15.10.3, "Package RAPL Domain."
611H	1553	MSR_PKG_ENERGY_STATUS	Package	PKG Energy Status (R/O)
				See Section 15.10.3, "Package RAPL Domain."
613H	1555	MSR_PKG_PERF_STATUS	Package	PKG Perf Status (R/O)
				See Section 15.10.3, "Package RAPL Domain."
614H	1556	MSR_PKG_POWER_INFO	Package	PKG RAPL Parameters (R/W)
				See Section 15.10.3, "Package RAPL Domain."
618H	1560	MSR_DRAM_POWER_LIMIT	Package	DRAM RAPL Power Limit Control (R/W)
				See Section 15.10.5, "DRAM RAPL Domain."
619H	1561	MSR_DRAM_ENERGY_STATUS	Package	DRAM Energy Status (R/O)
				See Section 15.10.5, "DRAM RAPL Domain."
61BH	1563	MSR_DRAM_PERF_STATUS	Package	DRAM Performance Throttling Status (R/O)
				See Section 15.10.5, "DRAM RAPL Domain."
61CH	1564	MSR_DRAM_POWER_INFO	Package	DRAM RAPL Parameters (R/W)
				See Section 15.10.5, "DRAM RAPL Domain."
638H	1592	MSR_PPO_POWER_LIMIT	Package	PPO RAPL Power Limit Control (R/W)
				See Section 15.10.4, "PPO/PP1 RAPL Domains."
639H	1593	MSR_PPO_ENERGY_STATUS	Package	PPO Energy Status (R/O)
				See Section 15.10.4, "PPO/PP1 RAPL Domains."
648H	1608	MSR_CONFIG_TDP_NOMINAL	Package	Base TDP Ratio (R/O)
				See Table 2-25.
649H	1609	MSR_CONFIG_TDP_LEVEL1	Package	ConfigTDP Level 1 ratio and power level (R/O) See Table 2-25.
64AH	1610	MSR_CONFIG_TDP_LEVEL2	Package	ConfigTDP Level 2 ratio and power level (R/O) See Table 2-25.

Table 2-53. Selected MSRs Supported by Intel® Xeon Phi™ Processors with DisplayFamily_DisplayModel Signatures 06_57H and 06_85H

_	ister ress	Register Name / Bit Fields (Former MSR Name)	Scope	Bit Description
Hex	Dec			
64BH	1611	MSR_CONFIG_TDP_CONTROL	Package	ConfigTDP Control (R/W) See Table 2-25.
64CH	1612	MSR_TURBO_ACTIVATION_RATIO	Package	ConfigTDP Control (R/W) See Table 2-25.
690H	1680	MSR_CORE_PERF_LIMIT_REASONS	Package	Indicator of Frequency Clipping in Processor Cores (R/W) (Frequency refers to processor core frequency.)
		0		PROCHOT Status (R0)
		1		Thermal Status (R0)
		5:2		Reserved
		6		VR Therm Alert Status (R0)
		7		Reserved
		8		Electrical Design Point Status (R0)
		63:9		Reserved
6E0H	1760	IA32_TSC_DEADLINE	Core	TSC Target of Local APIC's TSC Deadline Mode (R/W) See Table 2-2.
802H	2050	IA32_X2APIC_APICID	Thread	x2APIC ID Register (R/O)
803H	2051	IA32_X2APIC_VERSION	Thread	x2APIC Version Register (R/O)
808H	2056	IA32_X2APIC_TPR	Thread	x2APIC Task Priority Register (R/W)
HA08	2058	IA32_X2APIC_PPR	Thread	x2APIC Processor Priority Register (R/O)
80BH	2059	IA32_X2APIC_EOI	Thread	x2APIC EOI Register (W/O)
80DH	2061	IA32_X2APIC_LDR	Thread	x2APIC Logical Destination Register (R/0)
80FH	2063	IA32_X2APIC_SIVR	Thread	x2APIC Spurious Interrupt Vector Register (R/W)
810H	2064	IA32_X2APIC_ISR0	Thread	x2APIC In-Service Register Bits [31:0] (R/0)
811H	2065	IA32_X2APIC_ISR1	Thread	x2APIC In-Service Register Bits [63:32] (R/O)
812H	2066	IA32_X2APIC_ISR2	Thread	x2APIC In-Service Register Bits [95:64] (R/O)
813H	2067	IA32_X2APIC_ISR3	Thread	x2APIC In-Service Register Bits [127:96] (R/0)
814H	2068	IA32_X2APIC_ISR4	Thread	x2APIC In-Service Register Bits [159:128] (R/0)
815H	2069	IA32_X2APIC_ISR5	Thread	x2APIC In-Service Register Bits [191:160] (R/0)
816H	2070	IA32_X2APIC_ISR6	Thread	x2APIC In-Service Register Bits [223:192] (R/0)
817H	2071	IA32_X2APIC_ISR7	Thread	x2APIC In-Service Register Bits [255:224] (R/0)
818H	2072	IA32_X2APIC_TMR0	Thread	x2APIC Trigger Mode Register Bits [31:0] (R/O)
819H	2073	IA32_X2APIC_TMR1	Thread	x2APIC Trigger Mode Register Bits [63:32] (R/0)
81AH	2074	IA32_X2APIC_TMR2	Thread	x2APIC Trigger Mode Register Bits [95:64] (R/O)
81BH	2075	IA32_X2APIC_TMR3	Thread	x2APIC Trigger Mode Register Bits [127:96] (R/O)
81CH	2076	IA32_X2APIC_TMR4	Thread	x2APIC Trigger Mode Register Bits [159:128] (R/O)
81DH	2077	IA32_X2APIC_TMR5	Thread	x2APIC Trigger Mode Register Bits [191:160] (R/O)
81EH	2078	IA32_X2APIC_TMR6	Thread	x2APIC Trigger Mode Register Bits [223:192] (R/O)

Table 2-53. Selected MSRs Supported by Intel® Xeon Phi™ Processors with DisplayFamily_DisplayModel Signatures 06_57H and 06_85H

Register Address		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
81FH	2079	IA32_X2APIC_TMR7	Thread	x2APIC Trigger Mode Register Bits [255:224] (R/O)
820H	2080	IA32_X2APIC_IRRO	Thread	x2APIC Interrupt Request Register Bits [31:0] (R/0)
821H	2081	IA32_X2APIC_IRR1	Thread	x2APIC Interrupt Request Register Bits [63:32] (R/0)
822H	2082	IA32_X2APIC_IRR2	Thread	x2APIC Interrupt Request Register Bits [95:64] (R/0)
823H	2083	IA32_X2APIC_IRR3	Thread	x2APIC Interrupt Request Register Bits [127:96] (R/0)
824H	2084	IA32_X2APIC_IRR4	Thread	x2APIC Interrupt Request Register Bits [159:128] (R/0)
825H	2085	IA32_X2APIC_IRR5	Thread	x2APIC Interrupt Request Register Bits [191:160] (R/0)
826H	2086	IA32_X2APIC_IRR6	Thread	x2APIC Interrupt Request Register Bits [223:192] (R/0)
827H	2087	IA32_X2APIC_IRR7	Thread	x2APIC Interrupt Request Register Bits [255:224] (R/0)
828H	2088	IA32_X2APIC_ESR	Thread	x2APIC Error Status Register (R/W)
82FH	2095	IA32_X2APIC_LVT_CMCI	Thread	x2APIC LVT Corrected Machine Check Interrupt Register (R/W)
830H	2096	IA32_X2APIC_ICR	Thread	x2APIC Interrupt Command Register (R/W)
832H	2098	IA32_X2APIC_LVT_TIMER	Thread	x2APIC LVT Timer Interrupt Register (R/W)
833H	2099	IA32_X2APIC_LVT_THERMAL	Thread	x2APIC LVT Thermal Sensor Interrupt Register (R/W)
834H	2100	IA32_X2APIC_LVT_PMI	Thread	x2APIC LVT Performance Monitor Register (R/W)
835H	2101	IA32_X2APIC_LVT_LINTO	Thread	x2APIC LVT LINTO Register (R/W)
836H	2102	IA32_X2APIC_LVT_LINT1	Thread	x2APIC LVT LINT1 Register (R/W)
837H	2103	IA32_X2APIC_LVT_ERROR	Thread	x2APIC LVT Error Register (R/W)
838H	2104	IA32_X2APIC_INIT_COUNT	Thread	x2APIC Initial Count Register (R/W)
839H	2105	IA32_X2APIC_CUR_COUNT	Thread	x2APIC Current Count Register (R/O)
83EH	2110	IA32_X2APIC_DIV_CONF	Thread	x2APIC Divide Configuration Register (R/W)
83FH	2111	IA32_X2APIC_SELF_IPI	Thread	x2APIC Self IPI Register (W/O)
0080H		IA32_EFER	Thread	Extended Feature Enables See Table 2-2.
C000_ 0081H		IA32_STAR	Thread	System Call Target Address (R/W) See Table 2-2.
C000_ 0082H		IA32_LSTAR	Thread	IA-32e Mode System Call Target Address (R/W) See Table 2-2.
C000_ 0084H		IA32_FMASK	Thread	System Call Flag Mask (R/W) See Table 2-2.
C000_ 0100H		IA32_FS_BASE	Thread	Map of BASE Address of FS (R/W) See Table 2-2.
C000_ 0101H		IA32_GS_BASE	Thread	Map of BASE Address of GS (R/W) See Table 2-2.
C000_ 0102H		IA32_KERNEL_GS_BASE	Thread	Swap Target of BASE Address of GS (R/W) See Table 2-2.

Table 2-53. Selected MSRs Supported by Intel® Xeon Phi™ Processors with DisplayFamily_DisplayModel Signatures 06_57H and 06_85H

Regi Add		Register Name / Bit Fields	Scope	Bit Description
Hex	Dec	(Former MSR Name)		
C000_		IA32_TSC_AUX	Thread	AUXILIARY TSC Signature (R/W)
0103H				See Table 2-2

Table 2-54 lists model-specific registers that are supported by Intel[®] Xeon Phi[™] processor 7215, 7285, 7295 series based on the Knights Mill microarchitecture.

Table 2-54. Additional MSRs Supported by Intel® Xeon Phi™ Processor 7215, 7285, 7295 Series with DisplayFamily_DisplayModel Signature 06_85H

	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
9BH	155	IA32_SMM_MONITOR_CTL	Core	SMM Monitor Configuration (R/W) This MSR is readable only if VMX is enabled, and writeable only if VMX is enabled and in SMM mode, and is used to configure the VMX MSEG base address. See Table 2-2.
480H	1152	IA32_VMX_BASIC	Core	Reporting Register of Basic VMX Capabilities (R/O) See Table 2-2.
481H	1153	IA32_VMX_PINBASED_ CTLS	Core	Capability Reporting Register of Pin-based VM-execution Controls (R/O) See Table 2-2.
482H	1154	IA32_VMX_PROCBASED_ CTLS	Core	Capability Reporting Register of Primary Processor- based VM-execution Controls (R/O)
483H	1155	IA32_VMX_EXIT_CTLS	Core	Capability Reporting Register of VM-exit Controls (R/O) See Table 2-2.
484H	1156	IA32_VMX_ENTRY_CTLS	Core	Capability Reporting Register of VM-entry Controls (R/O) See Table 2-2.
485H	1157	IA32_VMX_MISC	Core	Reporting Register of Miscellaneous VMX Capabilities (R/O) See Table 2-2.
486H	1158	IA32_VMX_CR0_FIXED0	Core	Capability Reporting Register of CRO Bits Fixed to 0 (R/O) See Table 2-2.
487H	1159	IA32_VMX_CRO_FIXED1	Core	Capability Reporting Register of CRO Bits Fixed to 1 (R/O) See Table 2-2.
488H	1160	IA32_VMX_CR4_FIXED0	Core	Capability Reporting Register of CR4 Bits Fixed to 0 (R/0) See Table 2-2.
489H	1161	IA32_VMX_CR4_FIXED1	Core	Capability Reporting Register of CR4 Bits Fixed to 1 (R/0) See Table 2-2.

Table 2-54. Additional MSRs Supported by Intel® Xeon Phi™ Processor 7215, 7285, 7295 Series with DisplayFamily_DisplayModel Signature 06_85H

_	ister ress	Register Name / Bit Fields	Scope	Bit Description
Hex	Dec			
48AH	1162	IA32_VMX_VMCS_ENUM	Core	Capability Reporting Register of VMCS Field Enumeration (R/O) See Table 2-2.
48BH	1163	IA32_VMX_PROCBASED_ CTLS2	Core	Capability Reporting Register of Secondary Processor- Based VM-Execution Controls (R/O) See Table 2-2.
48CH	1164	IA32_VMX_EPT_VPID_ENUM	Core	Capability Reporting Register of EPT and VPID (R/O) See Table 2-2.
48DH	1165	IA32_VMX_TRUE_PINBASE D_CTLS	Core	Capability Reporting Register of Pin-Based VM-Execution Flex Controls (R/O) See Table 2-2.
48EH	1166	IA32_VMX_TRUE_PROCBASED_CTLS	Core	Capability Reporting Register of Primary Processor- Based VM-Execution Flex Controls (R/O) See Table 2-2.
48FH	1167	IA32_VMX_TRUE_EXIT_CTLS	Core	Capability Reporting Register of VM-Exit Flex Controls (R/O) See Table 2-2.
490H	1168	IA32_VMX_TRUE_ENTRY_CTLS	Core	Capability Reporting Register of VM-Entry Flex Controls (R/O) See Table 2-2.
491H	1169	IA32_VMX_FMFUNC	Core	Capability Reporting Register of VM-Function Controls (R/O) See Table 2-2.

2.19 MSRS IN THE PENTIUM® 4 AND INTEL® XEON® PROCESSORS

Table 2-55 lists MSRs (architectural and model-specific) that are defined across processor generations based on Intel NetBurst microarchitecture. The processor can be identified by its CPUID signatures of DisplayFamily encoding of 0FH, see Table 2-1.

- MSRs with an "IA32_" prefix are designated as "architectural." This means that the functions of these MSRs and their addresses remain the same for succeeding families of IA-32 processors.
- MSRs with an "MSR_" prefix are model specific with respect to address functionalities. The column "Model Availability" lists the model encoding value(s) within the Pentium 4 and Intel Xeon processor family at the specified register address. The model encoding value of a processor can be queried using CPUID. See "CPUID—CPU Identification" in Chapter 3 of the Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 2A.

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors

Regi Add	ister ress	Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
OH	0	IA32_P5_MC_ADDR	0, 1, 2, 3, 4, 6	Shared	See Section 2.23, "MSRs in Pentium Processors."

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Regi Add		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
1H	1	IA32_P5_MC_TYPE	0, 1, 2, 3, 4, 6	Shared	See Section 2.23, "MSRs in Pentium Processors."
6H	6	IA32_MONITOR_FILTER_LINE_SIZE	3, 4, 6	Shared	See Section 9.10.5, "Monitor/Mwait Address Range Determination."
10H	16	IA32_TIME_STAMP_COUNTER	0, 1, 2, 3, 4, 6	Unique	Time Stamp Counter See Table 2-2.
					On earlier processors, only the lower 32 bits are writable. On any write to the lower 32 bits, the upper 32 bits are cleared. For processor family OFH, models 3 and 4: all 64 bits are writable.
17H	23	IA32_PLATFORM_ID	0, 1, 2, 3, 4, 6	Shared	Platform ID (R) See Table 2-2. The operating system can use this MSR to determine "slot" information for the processor and the proper microcode update to load.
1BH	27	IA32_APIC_BASE	0, 1, 2, 3, 4, 6	Unique	APIC Location and Status (R/W) See Table 2-2. See Section 11.4.4, "Local APIC Status and Location."
2AH	42	MSR_EBC_HARD_POWERON	0, 1, 2, 3, 4, 6	Shared	Processor Hard Power-On Configuration (R/W) Enables and disables processor features. (R) Indicates current processor configuration.
		0			Output Tri-state Enabled (R) Indicates whether tri-state output is enabled (1) or disabled (0) as set by the strapping of SMI#. The value in this bit is written on the deassertion of RESET#; the bit is set to 1 when the address bus signal is asserted.
		1			Execute BIST (R) Indicates whether the execution of the BIST is enabled (1) or disabled (0) as set by the strapping of INIT#. The value in this bit is written on the deassertion of RESET#; the bit is set to 1 when the address bus signal is asserted.
		2			In Order Queue Depth (R) Indicates whether the in order queue depth for the system bus is 1 (1) or up to 12 (0) as set by the strapping of A7#. The value in this bit is written on the deassertion of RESET#; the bit is set to 1 when the address bus signal is asserted.

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Regi Addı		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
		3			MCERR# Observation Disabled (R) Indicates whether MCERR# observation is enabled (0) or disabled (1) as determined by the strapping of A9#. The value in this bit is written on the deassertion of RESET#; the bit is set to 1 when the address bus signal is asserted.
		4			BINIT# Observation Enabled (R) Indicates whether BINIT# observation is enabled (0) or disabled (1) as determined by the strapping of A10#. The value in this bit is written on the deassertion of RESET#; the bit is set to 1 when the address bus signal is asserted.
		6:5			APIC Cluster ID (R) Contains the logical APIC cluster ID value as set by the strapping of A12# and A11#. The logical cluster ID value is written into the field on the deassertion of RESET#; the field is set to 1 when the address bus signal is asserted.
		7			Bus Park Disable (R) Indicates whether bus park is enabled (0) or disabled (1) as set by the strapping of A15#. The value in this bit is written on the deassertion of RESET#; the bit is set to 1 when the address bus signal is asserted.
		11:8			Reserved
		13:12			Agent ID (R) Contains the logical agent ID value as set by the strapping of BR[3:0]. The logical ID value is written into the field on the deassertion of RESET#; the field is set to 1 when the address bus signal is asserted.
		63:14			Reserved
2BH	43	MSR_EBC_SOFT_POWERON 0	0, 1, 2, 3, 4, 6	Shared	Processor Soft Power-On Configuration (R/W) Enables and disables processor features. RCNT/SCNT On Request Encoding Enable (R/W)
		U			Controls the driving of RCNT/SCNT on the request encoding. Set to enable (1); clear to disabled (0, default).
		1			Data Error Checking Disable (R/W) Set to disable system data bus parity checking; clear to enable parity checking.
		2			Response Error Checking Disable (R/W) Set to disable (default); clear to enable.
		3			Address/Request Error Checking Disable (R/W) Set to disable (default); clear to enable.

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Regi Add		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
		4			Initiator MCERR# Disable (R/W) Set to disable MCERR# driving for initiator bus requests (default); clear to enable.
		5			Internal MCERR# Disable (R/W)
					Set to disable MCERR# driving for initiator internal errors (default); clear to enable.
		6			BINIT# Driver Disable (R/W)
					Set to disable BINIT# driver (default); clear to enable driver.
		63:7			Reserved
2CH	44	MSR_EBC_FREQUENCY_ID	2,3, 4, 6	Shared	Processor Frequency Configuration
					The bit field layout of this MSR varies according to the MODEL value in the CPUID version information. The following bit field layout applies to Pentium 4 and Xeon Processors with MODEL encoding equal or greater than 2. (R) The field Indicates the current processor
					frequency configuration.
		15:0			Reserved
		18:16			Scalable Bus Speed (R/W) Indicates the intended scalable bus speed:
					Encoding Scalable Bus Speed 000B 100 MHz (Model 2) 000B 266 MHz (Model 3 or 4) 001B 133 MHz 010B 200 MHz 011B 166 MHz 100B 333 MHz (Model 6)
					133.33 MHz should be utilized if performing calculation with System Bus Speed when encoding is 001B.
					166.67 MHz should be utilized if performing calculation with System Bus Speed when encoding is 011B.
					266.67 MHz should be utilized if performing calculation with System Bus Speed when encoding is 000B and model encoding = 3 or 4. 333.33 MHz should be utilized if performing
					calculation with System Bus Speed when encoding is 100B and model encoding = 6. All other values are reserved.
		23:19			Reserved

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Regi Add		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
		31:24			Core Clock Frequency to System Bus Frequency Ratio (R) The processor core clock frequency to system bus frequency ratio observed at the deassertion of the reset pin.
		63:25			Reserved
2CH	44	MSR_EBC_FREQUENCY_ID	0, 1	Shared	Processor Frequency Configuration (R)
					The bit field layout of this MSR varies according to the MODEL value of the CPUID version information. This bit field layout applies to Pentium 4 and Xeon Processors with MODEL encoding less than 2.
					Indicates current processor frequency configuration.
		20:0			Reserved
		23:21			Scalable Bus Speed (R/W)
					Indicates the intended scalable bus speed:
					Encoding Scalable Bus Speed 000B 100 MHz
					All others values reserved.
		63:24			Reserved
ЗАН	58	IA32_FEATURE_CONTROL	3, 4, 6	Unique	Control Features in IA-32 Processor (R/W) See Table 2-2. (If CPUID.01H:ECX.[bit 5])
79H	121	IA32_BIOS_UPDT_TRIG	0, 1, 2, 3, 4, 6	Shared	BIOS Update Trigger Register (W) See Table 2-2.
8BH	139	IA32_BIOS_SIGN_ID	0, 1, 2, 3, 4, 6	Unique	BIOS Update Signature ID (R/W) See Table 2-2.
9BH	155	IA32_SMM_MONITOR_CTL	3, 4, 6	Unique	SMM Monitor Configuration (R/W) See Table 2-2.
FEH	254	IA32_MTRRCAP	0, 1, 2, 3, 4, 6	Unique	MTRR Information See Section 12.11.1, "MTRR Feature Identification.".
174H	372	IA32_SYSENTER_CS	0, 1, 2, 3, 4, 6	Unique	CS Register Target for CPL 0 Code (R/W) See Table 2-2.
					See Section 5.8.7, "Performing Fast Calls to System Procedures with the SYSENTER and SYSEXIT Instructions."
175H	373	IA32_SYSENTER_ESP	0, 1, 2, 3, 4, 6	Unique	Stack Pointer for CPL 0 Stack (R/W) See Table 2-2.
					See Section 5.8.7, "Performing Fast Calls to System Procedures with the SYSENTER and SYSEXIT Instructions."

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Regi Addı		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
176H	374	IA32_SYSENTER_EIP	0, 1, 2, 3,	Unique	CPL 0 Code Entry Point (R/W)
			4, 6		See Table 2-2. See Section 5.8.7, "Performing Fast Calls to System Procedures with the SYSENTER and SYSEXIT Instructions."
179H	377	IA32_MCG_CAP	0, 1, 2, 3,	Unique	Machine Check Capabilities (R)
			4, 6		See Table 2-2. See Section 16.3.1.1, "IA32_MCG_CAP MSR."
17AH	378	IA32_MCG_STATUS	0, 1, 2, 3,	Unique	Machine Check Status (R)
			4, 6		See Table 2-2. See Section 16.3.1.2, "IA32_MCG_STATUS MSR."
17BH	379	IA32_MCG_CTL			Machine Check Feature Enable (R/W)
					See Table 2-2. See Section 16.3.1.3, "IA32_MCG_CTL MSR."
180H	384	MSR_MCG_RAX	0, 1, 2, 3,	Unique	Machine Check EAX/RAX Save State
1000	304	INSK_NCU_KAX	4, 6	Offique	See Section 16.3.2.6, "IA32_MCG Extended
					Machine Check State MSRs."
		63:0			Contains register state at time of machine check
					error. When in non-64-bit modes at the time of the error, bits 63-32 do not contain valid data.
181H	385	MSR_MCG_RBX	0, 1, 2, 3,	Unique	Machine Check EBX/RBX Save State
			4, 6		See Section 16.3.2.6, "IA32_MCG Extended Machine Check State MSRs."
		63:0			Contains register state at time of machine check
					error. When in non-64-bit modes at the time of the error, bits 63-32 do not contain valid data.
182H	386	MSR_MCG_RCX	0, 1, 2, 3,	Unique	Machine Check ECX/RCX Save State
			4, 6		See Section 16.3.2.6, "IA32_MCG Extended Machine Check State MSRs."
		63:0			Contains register state at time of machine check
					error. When in non-64-bit modes at the time of the error, bits 63-32 do not contain valid data.
183H	387	MSR_MCG_RDX	0, 1, 2, 3,	Unique	Machine Check EDX/RDX Save State
			4, 6		See Section 16.3.2.6, "IA32_MCG Extended Machine Check State MSRs."
		63:0			Contains register state at time of machine check
					error. When in non-64-bit modes at the time of the error, bits 63-32 do not contain valid data.
184H	388	MSR_MCG_RSI	0, 1, 2, 3,	Unique	Machine Check ESI/RSI Save State
			4, 6		See Section 16.3.2.6, "IA32_MCG Extended Machine Check State MSRs."
		63:0			Contains register state at time of machine check error. When in non-64-bit modes at the time of
					the error, bits 63-32 do not contain valid data.

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Regi Add		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
185H	389	MSR_MCG_RDI	0, 1, 2, 3, 4, 6	Unique	Machine Check EDI/RDI Save State See Section 16.3.2.6, "IA32_MCG Extended Machine Check State MSRs."
		63:0			Contains register state at time of machine check error. When in non-64-bit modes at the time of the error, bits 63-32 do not contain valid data.
186H	390	MSR_MCG_RBP	0, 1, 2, 3, 4, 6	Unique	Machine Check EBP/RBP Save State See Section 16.3.2.6, "IA32_MCG Extended Machine Check State MSRs."
		63:0			Contains register state at time of machine check error. When in non-64-bit modes at the time of the error, bits 63-32 do not contain valid data.
187H	391	MSR_MCG_RSP	0, 1, 2, 3,	Unique	Machine Check ESP/RSP Save State
			4, 6		See Section 16.3.2.6, "IA32_MCG Extended Machine Check State MSRs."
		63:0			Contains register state at time of machine check error. When in non-64-bit modes at the time of the error, bits 63-32 do not contain valid data.
188H	392	MSR_MCG_RFLAGS	0, 1, 2, 3,	Unique	Machine Check EFLAGS/RFLAG Save State
			4, 6		See Section 16.3.2.6, "IA32_MCG Extended Machine Check State MSRs."
		63:0			Contains register state at time of machine check error. When in non-64-bit modes at the time of the error, bits 63-32 do not contain valid data.
189H	393	MSR_MCG_RIP	0, 1, 2, 3,	Unique	Machine Check EIP/RIP Save State
			4, 6		See Section 16.3.2.6, "IA32_MCG Extended Machine Check State MSRs."
		63:0			Contains register state at time of machine check error. When in non-64-bit modes at the time of the error, bits 63-32 do not contain valid data.
18AH	394	MSR_MCG_MISC	0, 1, 2, 3,	Unique	Machine Check Miscellaneous
			4, 6		See Section 16.3.2.6, "IA32_MCG Extended Machine Check State MSRs."
		0			DS
					When set, the bit indicates that a page assist or page fault occurred during DS normal operation. The processors response is to shut down.
					The bit is used as an aid for debugging DS handling code. It is the responsibility of the user (BIOS or operating system) to clear this bit for normal operation.
		63:1			Reserved
18BH- 18FH	395- 399	MSR_MCG_RESERVED1 - MSR_MCG_RESERVED5			Reserved

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Regi Addı		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
190H	400	MSR_MCG_R8	0, 1, 2, 3, 4, 6	Unique	Machine Check R8 See Section 16.3.2.6, "IA32_MCG Extended Machine Check State MSRs."
		63:0			Registers R8-15 (and the associated state-save MSRs) exist only in Intel 64 processors. These registers contain valid information only when the processor is operating in 64-bit mode at the time of the error.
191H	401	MSR_MCG_R9	0, 1, 2, 3,	Unique	Machine Check R9D/R9
			4, 6		See Section 16.3.2.6, "IA32_MCG Extended Machine Check State MSRs."
		63:0			Registers R8-15 (and the associated state-save MSRs) exist only in Intel 64 processors. These registers contain valid information only when the processor is operating in 64-bit mode at the time of the error.
192H	402	MSR_MCG_R10	0, 1, 2, 3,	Unique	Machine Check R10
			4, 6		See Section 16.3.2.6, "IA32_MCG Extended Machine Check State MSRs."
		63:0			Registers R8-15 (and the associated state-save MSRs) exist only in Intel 64 processors. These registers contain valid information only when the processor is operating in 64-bit mode at the time of the error.
193H	403	MSR_MCG_R11	0, 1, 2, 3,	Unique	Machine Check R11
			4, 6		See Section 16.3.2.6, "IA32_MCG Extended Machine Check State MSRs."
		63:0			Registers R8-15 (and the associated state-save MSRs) exist only in Intel 64 processors. These registers contain valid information only when the processor is operating in 64-bit mode at the time of the error.
194H	404	MSR_MCG_R12	0, 1, 2, 3,	Unique	Machine Check R12
			4, 6		See Section 16.3.2.6, "IA32_MCG Extended Machine Check State MSRs."
		63:0			Registers R8-15 (and the associated state-save MSRs) exist only in Intel 64 processors. These registers contain valid information only when the processor is operating in 64-bit mode at the time of the error.
195H	405	MSR_MCG_R13	0, 1, 2, 3,	Unique	Machine Check R13
			4, 6		See Section 16.3.2.6, "IA32_MCG Extended Machine Check State MSRs."

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Regi Add		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
		63:0			Registers R8-15 (and the associated state-save MSRs) exist only in Intel 64 processors. These registers contain valid information only when the processor is operating in 64-bit mode at the time of the error.
196H	406	MSR_MCG_R14	0, 1, 2, 3, 4, 6	Unique	Machine Check R14 See Section 16.3.2.6, "IA32_MCG Extended Machine Check State MSRs."
		63:0			Registers R8-15 (and the associated state-save MSRs) exist only in Intel 64 processors. These registers contain valid information only when the processor is operating in 64-bit mode at the time of the error.
197H	407	MSR_MCG_R15	0, 1, 2, 3, 4, 6	Unique	Machine Check R15
			4,0		See Section 16.3.2.6, "IA32_MCG Extended Machine Check State MSRs."
		63:0			Registers R8-15 (and the associated state-save MSRs) exist only in Intel 64 processors. These registers contain valid information only when the processor is operating in 64-bit mode at the time of the error.
198H	408	IA32_PERF_STATUS	3, 4, 6	Unique	See Table 2-2. See Section 15.1, "Enhanced Intel Speedstep [®] Technology."
199H	409	IA32_PERF_CTL	3, 4, 6	Unique	See Table 2-2. See Section 15.1, "Enhanced Intel Speedstep [®] Technology."
19AH	410	IA32_CLOCK_MODULATION	0, 1, 2, 3, 4, 6	Unique	Thermal Monitor Control (R/W) See Table 2-2. See Section 15.8.3, "Software Controlled Clock Modulation."
19BH	411	IA32_THERM_INTERRUPT	0, 1, 2, 3, 4, 6	Unique	Thermal Interrupt Control (R/W) See Section 15.8.2, "Thermal Monitor," and see Table 2-2.
19CH	412	IA32_THERM_STATUS	0, 1, 2, 3, 4, 6	Shared	Thermal Monitor Status (R/W) See Section 15.8.2, "Thermal Monitor," and see Table 2-2.
19DH	413	MSR_THERM2_CTL			Thermal Monitor 2 Control
			3,	Shared	For Family F, Model 3 processors: When read, specifies the value of the target TM2 transition last written. When set, it sets the next target value for TM2 transition.
			4, 6	Shared	For Family F, Model 4 and Model 6 processors: When read, specifies the value of the target TM2 transition last written. Writes may cause #GP exceptions.
1A0H	416	IA32_MISC_ENABLE	0, 1, 2, 3, 4, 6	Shared	Enable Miscellaneous Processor Features (R/W)

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Regi: Addr		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
		0			Fast-Strings Enable. See Table 2-2.
		1			Reserved
		2			x87 FPU Fopcode Compatibility Mode Enable
		3			Thermal Monitor 1 Enable
					See Section 15.8.2, "Thermal Monitor," and see Table 2-2.
		4			Split-Lock Disable
					When set, the bit causes an #AC exception to be issued instead of a split-lock cycle. Operating systems that set this bit must align system structures to avoid split-lock scenarios. When the bit is clear (default), normal split-locks are issued to the bus.
					This debug feature is specific to the Pentium 4 processor.
		5			Reserved
		6			Third-Level Cache Disable (R/W)
					When set, the third-level cache is disabled; when clear (default) the third-level cache is enabled. This flag is reserved for processors that do not have a third-level cache.
					Note that the bit controls only the third-level cache; and only if overall caching is enabled through the CD flag of control register CRO, the page-level cache controls, and/or the MTRRs.
					See Section 12.5.4, "Disabling and Enabling the L3 Cache."
		7			Performance Monitoring Available (R)
					See Table 2-2.
		8			Suppress Lock Enable
					When set, assertion of LOCK on the bus is suppressed during a Split Lock access. When clear (default), LOCK is not suppressed.
		9			Prefetch Queue Disable
					When set, disables the prefetch queue. When clear (default), enables the prefetch queue.
		10			FERR# Interrupt Reporting Enable (R/W)
					When set, interrupt reporting through the FERR# pin is enabled; when clear, this interrupt reporting function is disabled.

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Register Address		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
					When this flag is set and the processor is in the stop-clock state (STPCLK# is asserted), asserting the FERR# pin signals to the processor that an interrupt (such as, INIT#, BINIT#, INTR, NMI, SMI#, or RESET#) is pending and that the processor should return to normal operation to handle the interrupt.
					This flag does not affect the normal operation of the FERR# pin (to indicate an unmasked floating-point error) when the STPCLK# pin is not asserted.
		11			Branch Trace Storage Unavailable (BTS_UNAVILABLE) (R) See Table 2-2.
					When set, the processor does not support branch trace storage (BTS); when clear, BTS is supported.
		12			PEBS_UNAVILABLE: Processor Event Based Sampling Unavailable (R) See Table 2-2.
					When set, the processor does not support processor event-based sampling (PEBS); when clear, PEBS is supported.
		13	3		TM2 Enable (R/W)
					When this bit is set (1) and the thermal sensor indicates that the die temperature is at the predetermined threshold, the Thermal Monitor 2 mechanism is engaged. TM2 will reduce the bus to core ratio and voltage according to the value last written to MSR_THERM2_CTL bits 15:0.
					When this bit is clear (0, default), the processor does not change the VID signals or the bus to core ratio when the processor enters a thermal managed state.
					If the TM2 feature flag (ECX[8]) is not set to 1 after executing CPUID with EAX = 1, then this feature is not supported and BIOS must not alter the contents of this bit location. The processor is operating out of spec if both this bit and the TM1 bit are set to disabled states.
		17:14			Reserved
		18	3, 4, 6		ENABLE MONITOR FSM (R/W)
					See Table 2-2.

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Register Address		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
		19			Adjacent Cache Line Prefetch Disable (R/W) When set to 1, the processor fetches the cache line of the 128-byte sector containing currently required data. When set to 0, the processor fetches both cache lines in the sector.
					Single processor platforms should not set this bit. Server platforms should set or clear this bit based on platform performance observed in validation and testing. BIOS may contain a setup option that controls the setting of this bit.
		21:20			Reserved
		22	3, 4, 6		Limit CPUID MAXVAL (R/W)
			3, 4, 0		See Table 2-2.
					Setting this can cause unexpected behavior to software that depends on the availability of CPUID leaves greater than 3.
		23		Shared	xTPR Message Disable (R/W)
					See Table 2-2.
		24			L1 Data Cache Context Mode (R/W)
					When set, the L1 data cache is placed in shared mode; when clear (default), the cache is placed in adaptive mode. This bit is only enabled for IA-32 processors that support Intel Hyper-Threading Technology. See Section 12.5.6, "L1 Data Cache Context Mode."
					When L1 is running in adaptive mode and CR3s are identical, data in L1 is shared across logical processors. Otherwise, L1 is not shared and cache use is competitive.
					If the Context ID feature flag (ECX[10]) is set to 0 after executing CPUID with EAX = 1, the ability to switch modes is not supported. BIOS must not alter the contents of IA32_MISC_ENABLE[24].
		33:25			Reserved
		34		Unique	XD Bit Disable (R/W)
					See Table 2-2.
		63:35			Reserved
1A1H	417	MSR_PLATFORM_BRV	3, 4, 6	Shared	Platform Feature Requirements (R)
		17:0			Reserved

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Regi Add		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
		18			PLATFORM Requirements When set to 1, indicates the processor has specific platform requirements. The details of the platform requirements are listed in the respective data sheets of the processor.
		63:19			Reserved
1D7H	471	MSR_LER_FROM_LIP	0, 1, 2, 3, 4, 6	Unique	Last Exception Record From Linear IP (R) Contains a pointer to the last branch instruction that the processor executed prior to the last exception that was generated or the last interrupt that was handled. See Section 18.13.3, "Last Exception Records."
		31:0			From Linear IP
					Linear address of the last branch instruction.
		63:32			Reserved
1D7H	471	63:0		Unique	From Linear IP
					Linear address of the last branch instruction (If IA-32e mode is active).
1D8H	472	MSR_LER_TO_LIP	0, 1, 2, 3,	Unique	Last Exception Record To Linear IP (R)
			4, 6		This area contains a pointer to the target of the last branch instruction that the processor executed prior to the last exception that was generated or the last interrupt that was handled. See Section 18.13.3, "Last Exception Records."
		31:0			From Linear IP
					Linear address of the target of the last branch instruction.
		63:32			Reserved
1D8H	472	63:0		Unique	From Linear IP
					Linear address of the target of the last branch instruction (If IA-32e mode is active).
1D9H	473	MSR_DEBUGCTLA	0, 1, 2, 3,	Unique	Debug Control (R/W)
			4, 6		Controls how several debug features are used. Bit definitions are discussed in the referenced section.
45	4	MOD LAGTED ASSESSED	0.5.5.5	1	See Section 18.13.1, "MSR_DEBUGCTLA MSR."
1DAH	474	MSR_LASTBRANCH_TOS	0, 1, 2, 3, 4, 6	Unique	Last Branch Record Stack TOS (R/O) Contains an index (0-3 or 0-15) that points to the top of the last branch record stack (that is, that points the index of the MSR containing the most recent branch record). See Section 18.13.2, "LBR Stack for Processors
					Based on Intel NetBurst® Microarchitecture"; and addresses 1DBH-1DEH and 680H-68FH.

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Regi Addı		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
1DBH	475	MSR_LASTBRANCH_0	0, 1, 2	Unique	Last Branch Record 0 (R/O) One of four last branch record registers on the last branch record stack. It contains pointers to the source and destination instruction for one of the last four branches, exceptions, or interrupts that the processor took. MSR_LASTBRANCH_O through MSR_LASTBRANCH_3 at 1DBH-1DEH are available only on family 0FH, models 0H-02H. They have been replaced by the MSRs at 680H-68FH and 6COH-6CFH.
					See Section 18.12, "Last Branch, Call Stack, Interrupt, and Exception Recording for Processors based on Skylake Microarchitecture."
1DCH	477	MSR_LASTBRANCH_1	0, 1, 2	Unique	Last Branch Record 1 See description of the MSR_LASTBRANCH_0 MSR at 1DBH.
1DDH	477	MSR_LASTBRANCH_2	0, 1, 2	Unique	Last Branch Record 2 See description of the MSR_LASTBRANCH_0 MSR at 1DBH.
1DEH	478	MSR_LASTBRANCH_3	0, 1, 2	Unique	Last Branch Record 3 See description of the MSR_LASTBRANCH_0 MSR at 1DBH.
200H	512	IA32_MTRR_PHYSBASE0	0, 1, 2, 3, 4, 6	Shared	Variable Range Base MTRR See Section 12.11.2.3, "Variable Range MTRRs."
201H	513	IA32_MTRR_PHYSMASKO	0, 1, 2, 3, 4, 6	Shared	Variable Range Mask MTRR See Section 12.11.2.3, "Variable Range MTRRs."
202H	514	IA32_MTRR_PHYSBASE1	0, 1, 2, 3, 4, 6	Shared	Variable Range Mask MTRR See Section 12.11.2.3, "Variable Range MTRRs."
203H	515	IA32_MTRR_PHYSMASK1	0, 1, 2, 3, 4, 6	Shared	Variable Range Mask MTRR See Section 12.11.2.3, "Variable Range MTRRs."
204H	516	IA32_MTRR_PHYSBASE2	0, 1, 2, 3, 4, 6	Shared	Variable Range Mask MTRR See Section 12.11.2.3, "Variable Range MTRRs."
205H	517	IA32_MTRR_PHYSMASK2	0, 1, 2, 3, 4, 6	Shared	Variable Range Mask MTRR See Section 12.11.2.3, "Variable Range MTRRs".
206H	518	IA32_MTRR_PHYSBASE3	0, 1, 2, 3, 4, 6	Shared	Variable Range Mask MTRR See Section 12.11.2.3, "Variable Range MTRRs."
207H	519	IA32_MTRR_PHYSMASK3	0, 1, 2, 3, 4, 6	Shared	Variable Range Mask MTRR See Section 12.11.2.3, "Variable Range MTRRs."
208H	520	IA32_MTRR_PHYSBASE4	0, 1, 2, 3, 4, 6	Shared	Variable Range Mask MTRR See Section 12.11.2.3, "Variable Range MTRRs."
209H	521	IA32_MTRR_PHYSMASK4	0, 1, 2, 3, 4, 6	Shared	Variable Range Mask MTRR See Section 12.11.2.3, "Variable Range MTRRs."

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Regi Add		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
20AH	522	IA32_MTRR_PHYSBASE5	0, 1, 2, 3, 4, 6	Shared	Variable Range Mask MTRR See Section 12.11.2.3, "Variable Range MTRRs."
20BH	523	IA32_MTRR_PHYSMASK5	0, 1, 2, 3, 4, 6	Shared	Variable Range Mask MTRR See Section 12.11.2.3, "Variable Range MTRRs."
20CH	524	IA32_MTRR_PHYSBASE6	0, 1, 2, 3, 4, 6	Shared	Variable Range Mask MTRR See Section 12.11.2.3, "Variable Range MTRRs."
20DH	525	IA32_MTRR_PHYSMASK6	0, 1, 2, 3, 4, 6	Shared	Variable Range Mask MTRR See Section 12.11.2.3, "Variable Range MTRRs."
20EH	526	IA32_MTRR_PHYSBASE7	0, 1, 2, 3, 4, 6	Shared	Variable Range Mask MTRR See Section 12.11.2.3, "Variable Range MTRRs."
20FH	527	IA32_MTRR_PHYSMASK7	0, 1, 2, 3, 4, 6	Shared	Variable Range Mask MTRR See Section 12.11.2.3, "Variable Range MTRRs."
250H	592	IA32_MTRR_FIX64K_00000	0, 1, 2, 3, 4, 6	Shared	Fixed Range MTRR See Section 12.11.2.2, "Fixed Range MTRRs."
258H	600	IA32_MTRR_FIX16K_80000	0, 1, 2, 3, 4, 6	Shared	Fixed Range MTRR See Section 12.11.2.2, "Fixed Range MTRRs."
259H	601	IA32_MTRR_FIX16K_A0000	0, 1, 2, 3, 4, 6	Shared	Fixed Range MTRR See Section 12.11.2.2, "Fixed Range MTRRs."
268H	616	IA32_MTRR_FIX4K_C0000	0, 1, 2, 3, 4, 6	Shared	Fixed Range MTRR See Section 12.11.2.2, "Fixed Range MTRRs."
269H	617	IA32_MTRR_FIX4K_C8000	0, 1, 2, 3, 4, 6	Shared	Fixed Range MTRR See Section 12.11.2.2, "Fixed Range MTRRs".
26AH	618	IA32_MTRR_FIX4K_D0000	0, 1, 2, 3, 4, 6	Shared	Fixed Range MTRR See Section 12.11.2.2, "Fixed Range MTRRs".
26BH	619	IA32_MTRR_FIX4K_D8000	0, 1, 2, 3, 4, 6	Shared	Fixed Range MTRR See Section 12.11.2.2, "Fixed Range MTRRs."
26CH	620	IA32_MTRR_FIX4K_E0000	0, 1, 2, 3, 4, 6	Shared	Fixed Range MTRR See Section 12.11.2.2, "Fixed Range MTRRs."
26DH	621	IA32_MTRR_FIX4K_E8000	0, 1, 2, 3, 4, 6	Shared	Fixed Range MTRR See Section 12.11.2.2, "Fixed Range MTRRs."
26EH	622	IA32_MTRR_FIX4K_F0000	0, 1, 2, 3, 4, 6	Shared	Fixed Range MTRR See Section 12.11.2.2, "Fixed Range MTRRs."
26FH	623	IA32_MTRR_FIX4K_F8000	0, 1, 2, 3, 4, 6	Shared	Fixed Range MTRR See Section 12.11.2.2, "Fixed Range MTRRs."
277H	631	IA32_PAT	0, 1, 2, 3, 4, 6	Unique	Page Attribute Table See Section 12.11.2.2, "Fixed Range MTRRs."
2FFH	767	IA32_MTRR_DEF_TYPE	0, 1, 2, 3, 4, 6	Shared	Default Memory Types (R/W) See Table 2-2. See Section 12.11.2.1, "IA32_MTRR_DEF_TYPE MSR."

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Regi Addı		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
300H	768	MSR_BPU_COUNTER0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.2, "Performance Counters."
301H	769	MSR_BPU_COUNTER1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.2, "Performance Counters."
302H	770	MSR_BPU_COUNTER2	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.2, "Performance Counters."
303H	771	MSR_BPU_COUNTER3	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.2, "Performance Counters."
304H	772	MSR_MS_COUNTERO	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.2, "Performance Counters."
305H	773	MSR_MS_COUNTER1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.2, "Performance Counters."
306H	774	MSR_MS_COUNTER2	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.2, "Performance Counters."
307H	775	MSR_MS_COUNTER3	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.2, "Performance Counters."
308H	776	MSR_FLAME_COUNTERO	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.2, "Performance Counters."
309H	777	MSR_FLAME_COUNTER1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.2, "Performance Counters."
30AH	778	MSR_FLAME_COUNTER2	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.2, "Performance Counters."
30BH	779	MSR_FLAME_COUNTER3	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.2, "Performance Counters."
30CH	780	MSR_IQ_COUNTERO	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.2, "Performance Counters."
30DH	781	MSR_IQ_COUNTER1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.2, "Performance Counters."
30EH	782	MSR_IQ_COUNTER2	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.2, "Performance Counters."
30FH	783	MSR_IQ_COUNTER3	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.2, "Performance Counters."
310H	784	MSR_IQ_COUNTER4	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.2, "Performance Counters."
311H	785	MSR_IQ_COUNTER5	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.2, "Performance Counters."
360H	864	MSR_BPU_CCCRO	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.3, "CCCR MSRs."
361H	865	MSR_BPU_CCCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.3, "CCCR MSRs."
362H	866	MSR_BPU_CCCR2	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.3, "CCCR MSRs."
363H	867	MSR_BPU_CCCR3	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.3, "CCCR MSRs."

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Regi Add		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
364H	868	MSR_MS_CCCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.3, "CCCR MSRs."
365H	869	MSR_MS_CCCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.3, "CCCR MSRs."
366H	870	MSR_MS_CCCR2	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.3, "CCCR MSRs."
367H	871	MSR_MS_CCCR3	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.3, "CCCR MSRs."
368H	872	MSR_FLAME_CCCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.3, "CCCR MSRs."
369H	873	MSR_FLAME_CCCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.3, "CCCR MSRs."
36AH	874	MSR_FLAME_CCCR2	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.3, "CCCR MSRs."
36BH	875	MSR_FLAME_CCCR3	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.3, "CCCR MSRs."
36CH	876	MSR_IQ_CCCRO	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.3, "CCCR MSRs."
36DH	877	MSR_IQ_CCCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.3, "CCCR MSRs."
36EH	878	MSR_IQ_CCCR2	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.3, "CCCR MSRs."
36FH	879	MSR_IQ_CCCR3	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.3, "CCCR MSRs."
370H	880	MSR_IQ_CCCR4	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.3, "CCCR MSRs."
371H	881	MSR_IQ_CCCR5	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.3, "CCCR MSRs."
3A0H	928	MSR_BSU_ESCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3A1H	929	MSR_BSU_ESCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
ЗА2Н	930	MSR_FSB_ESCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
ЗАЗН	931	MSR_FSB_ESCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3A4H	932	MSR_FIRM_ESCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3A5H	933	MSR_FIRM_ESCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3A6H	934	MSR_FLAME_ESCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
ЗА7Н	935	MSR_FLAME_ESCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Regi Addı		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
3A8H	936	MSR_DAC_ESCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3A9H	937	MSR_DAC_ESCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
ЗААН	938	MSR_MOB_ESCRO	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
ЗАВН	939	MSR_MOB_ESCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3ACH	940	MSR_PMH_ESCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3ADH	941	MSR_PMH_ESCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3AEH	942	MSR_SAAT_ESCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3AFH	943	MSR_SAAT_ESCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3B0H	944	MSR_U2L_ESCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3B1H	945	MSR_U2L_ESCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3B2H	946	MSR_BPU_ESCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3B3H	947	MSR_BPU_ESCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3B4H	948	MSR_IS_ESCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3B5H	949	MSR_IS_ESCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3B6H	950	MSR_ITLB_ESCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3B7H	951	MSR_ITLB_ESCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3B8H	952	MSR_CRU_ESCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3B9H	953	MSR_CRU_ESCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
ЗВАН	954	MSR_IQ_ESCR0	0, 1, 2	Shared	See Section 20.6.3.1, "ESCR MSRs."
					This MSR is not available on later processors. It is only available on processor family 0FH, models 01H-02H.
3BBH	955	MSR_IQ_ESCR1	0, 1, 2	Shared	See Section 20.6.3.1, "ESCR MSRs."
					This MSR is not available on later processors. It is only available on processor family 0FH, models 01H-02H.

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Regi Addi		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
ЗВСН	956	MSR_RAT_ESCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3BDH	957	MSR_RAT_ESCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3BEH	958	MSR_SSU_ESCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3C0H	960	MSR_MS_ESCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3C1H	961	MSR_MS_ESCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3C2H	962	MSR_TBPU_ESCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3C3H	963	MSR_TBPU_ESCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3C4H	964	MSR_TC_ESCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3C5H	965	MSR_TC_ESCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3C8H	968	MSR_IX_ESCR0	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3C9H	969	MSR_IX_ESCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3CAH	970	MSR_ALF_ESCRO	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3CBH	971	MSR_ALF_ESCR1	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3CCH	972	MSR_CRU_ESCR2	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3CDH	973	MSR_CRU_ESCR3	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3E0H	992	MSR_CRU_ESCR4	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3E1H	993	MSR_CRU_ESCR5	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3F0H	1008	MSR_TC_PRECISE_EVENT	0, 1, 2, 3, 4, 6	Shared	See Section 20.6.3.1, "ESCR MSRs."
3F1H	1009	IA32_PEBS_ENABLE (MSR_PEBS_ENABLE)	0, 1, 2, 3, 4, 6	Shared	Processor Event Based Sampling (PEBS) (R/W) Controls the enabling of processor event sampling and replay tagging.
		12:0			See https://perfmon-events.intel.com/.
		23:13			Reserved
		24			UOP Tag Enables replay tagging when set.

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Regi Addı		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
		25			ENABLE_PEBS_MY_THR (R/W) Enables PEBS for the target logical processor when set; disables PEBS when clear (default). See Section 20.6.4.3, "IA32_PEBS_ENABLE MSR," for an explanation of the target logical processor. This bit is called ENABLE_PEBS in IA-32 processors that do not support Intel Hyper-Threading Technology.
		26			ENABLE_PEBS_OTH_THR (R/W)
					Enables PEBS for the target logical processor when set; disables PEBS when clear (default).
					See Section 20.6.4.3, "IA32_PEBS_ENABLE MSR," for an explanation of the target logical processor.
					This bit is reserved for IA-32 processors that do not support Intel Hyper-Threading Technology.
		63:27			Reserved
3F2H	1010	MSR_PEBS_MATRIX_VERT	0, 1, 2, 3, 4, 6	Shared	See https://perfmon-events.intel.com/ .
400H	1024	IA32_MCO_CTL	0, 1, 2, 3, 4, 6	Shared	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
401H	1025	IA32_MCO_STATUS	0, 1, 2, 3, 4, 6	Shared	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
402H	1026	IA32_MCO_ADDR	0, 1, 2, 3, 4, 6	Shared	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs." The IA32_MCO_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MCO_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
403H	1027	IA32_MC0_MISC	0, 1, 2, 3, 4, 6	Shared	See Section 16.3.2.4, "IA32_MCi_MISC MSRs." The IA32_MCO_MISC MSR is either not implemented or does not contain additional information if the MISCV flag in the IA32_MCO_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
404H	1028	IA32_MC1_CTL	0, 1, 2, 3, 4, 6	Shared	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
405H	1029	IA32_MC1_STATUS	0, 1, 2, 3, 4, 6	Shared	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Regi Add		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
406H	1030	IA32_MC1_ADDR	0, 1, 2, 3, 4, 6	Shared	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs." The IA32_MC1_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MC1_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
407H	1031	IA32_MC1_MISC		Shared	See Section 16.3.2.4, "IA32_MCi_MISC MSRs." The IA32_MC1_MISC MSR is either not implemented or does not contain additional information if the MISCV flag in the IA32_MC1_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
408H	1032	IA32_MC2_CTL	0, 1, 2, 3, 4, 6	Shared	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
409H	1033	IA32_MC2_STATUS	0, 1, 2, 3, 4, 6	Shared	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
40AH	1034	IA32_MC2_ADDR			See Section 16.3.2.3, "IA32_MCi_ADDR MSRs." The IA32_MC2_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MC2_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
40BH	1035	IA32_MC2_MISC			See Section 16.3.2.4, "IA32_MCi_MISC MSRs." The IA32_MC2_MISC MSR is either not implemented or does not contain additional information if the MISCV flag in the IA32_MC2_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
40CH	1036	IA32_MC3_CTL	0, 1, 2, 3, 4, 6	Shared	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
40DH	1037	IA32_MC3_STATUS	0, 1, 2, 3, 4, 6	Shared	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
40EH	1038	IA32_MC3_ADDR	0, 1, 2, 3, 4, 6	Shared	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs." The IA32_MC3_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MC3_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Regi Add		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
40FH	1039	IA32_MC3_MISC	0, 1, 2, 3, 4, 6	Shared	See Section 16.3.2.4, "IA32_MCi_MISC MSRs." The IA32_MC3_MISC MSR is either not implemented or does not contain additional information if the MISCV flag in the IA32_MC3_STATUS register is clear.
					When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
410H	1040	IA32_MC4_CTL	0, 1, 2, 3, 4, 6	Shared	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
411H	1041	IA32_MC4_STATUS	0, 1, 2, 3, 4, 6	Shared	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
412H	1042	IA32_MC4_ADDR			See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
					The IA32_MC2_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MC4_STATUS register is clear.
					When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
413H	1043	IA32_MC4_MISC			See Section 16.3.2.4, "IA32_MCi_MISC MSRs."
					The IA32_MC2_MISC MSR is either not implemented or does not contain additional information if the MISCV flag in the IA32_MC4_STATUS register is clear.
					When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
480H	1152	IA32_VMX_BASIC	3, 4, 6	Unique	Reporting Register of Basic VMX Capabilities (R/O)
					See Table 2-2.
481H	1153	IA32_VMX_PINBASED_CTLS	3, 4, 6	Unique	See Appendix A.1, "Basic VMX Information." Capability Reporting Register of Pin-Based
40111	1133	INDE_VIMA_MINDADED_CTED	3, 4 , 0	Unique	VM-Execution Controls (R/O)
					See Table 2-2.
					See Appendix A.3, "VM-Execution Controls."
482H	1154	IA32_VMX_PROCBASED_CTLS	3, 4, 6	Unique	Capability Reporting Register of Primary Processor-Based VM-Execution Controls (R/O)
					See Appendix A.3, "VM-Execution Controls," and see Table 2-2.
483H	1155	IA32_VMX_EXIT_CTLS	3, 4, 6	Unique	Capability Reporting Register of VM-Exit Controls (R/O)
					See Appendix A.4, "VM-Exit Controls," and see Table 2-2.

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

	ister ress	Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
484H	1156	IA32_VMX_ENTRY_CTLS	3, 4, 6	Unique	Capability Reporting Register of VM-Entry Controls (R/O)
					See Appendix A.5, "VM-Entry Controls," and see Table 2-2.
485H	1157	IA32_VMX_MISC	3, 4, 6	Unique	Reporting Register of Miscellaneous VMX Capabilities (R/O)
					See Appendix A.6, "Miscellaneous Data," and see Table 2-2.
486H	1158	IA32_VMX_CRO_FIXEDO	3, 4, 6	Unique	Capability Reporting Register of CRO Bits Fixed to 0 (R/O)
					See Appendix A.7, "VMX-Fixed Bits in CR0," and see Table 2-2.
487H	1159	IA32_VMX_CRO_FIXED1	3, 4, 6	Unique	Capability Reporting Register of CRO Bits Fixed to 1 (R/O)
					See Appendix A.7, "VMX-Fixed Bits in CR0," and see Table 2-2.
488H	1160	IA32_VMX_CR4_FIXED0	3, 4, 6	Unique	Capability Reporting Register of CR4 Bits Fixed to 0 (R/O)
					See Appendix A.8, "VMX-Fixed Bits in CR4," and see Table 2-2.
489H	1161	IA32_VMX_CR4_FIXED1	3, 4, 6	Unique	Capability Reporting Register of CR4 Bits Fixed to 1 (R/O)
					See Appendix A.8, "VMX-Fixed Bits in CR4," and see Table 2-2.
48AH	1162	IA32_VMX_VMCS_ENUM	3, 4, 6	Unique	Capability Reporting Register of VMCS Field Enumeration (R/O)
					See Appendix A.9, "VMCS Enumeration," and see Table 2-2.
48BH	1163	IA32_VMX_PROCBASED_CTLS2	3, 4, 6	Unique	Capability Reporting Register of Secondary Processor-Based VM-Execution Controls (R/O)
					See Appendix A.3, "VM-Execution Controls," and see Table 2-2.
600H	1536	IA32_DS_AREA	0, 1, 2, 3,	Unique	DS Save Area (R/W)
			4, 6		See Table 2-2.
					See Section 20.6.3.4, "Debug Store (DS) Mechanism."
680H	1664	MSR_LASTBRANCH_0_FROM_IP	3, 4, 6	Unique	Last Branch Record 0 (R/W)
					One of 16 pairs of last branch record registers on the last branch record stack (680H-68FH). This part of the stack contains pointers to the source instruction for one of the last 16 branches, exceptions, or interrupts taken by the
					branches, exceptions, or interrupts taken to processor.

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

_	ister ress	Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
					The MSRs at 680H-68FH, 6C0H-6CfH are not available in processor releases before family 0FH, model 03H. These MSRs replace MSRs previously located at 1DBH-1DEH.which performed the same function for early releases.
					See Section 18.12, "Last Branch, Call Stack, Interrupt, and Exception Recording for Processors based on Skylake Microarchitecture."
681H	1665	MSR_LASTBRANCH_1_FROM_IP	3, 4, 6	Unique	Last Branch Record 1
					See description of MSR_LASTBRANCH_0 at 680H.
682H	1666	MSR_LASTBRANCH_2_FROM_IP	3, 4, 6	Unique	Last Branch Record 2
					See description of MSR_LASTBRANCH_0 at 680H.
683H	1667	MSR_LASTBRANCH_3_FROM_IP	3, 4, 6	Unique	Last Branch Record 3
					See description of MSR_LASTBRANCH_0 at 680H.
684H	1668	MSR_LASTBRANCH_4_FROM_IP	3, 4, 6	Unique	Last Branch Record 4
					See description of MSR_LASTBRANCH_0 at 680H.
685H	1669	MSR_LASTBRANCH_5_FROM_IP	3, 4, 6	Unique	Last Branch Record 5
					See description of MSR_LASTBRANCH_0 at 680H.
686H	1670	MSR_LASTBRANCH_6_FROM_IP	3, 4, 6	Unique	Last Branch Record 6
					See description of MSR_LASTBRANCH_0 at 680H.
687H	1671	MSR_LASTBRANCH_7_FROM_IP	3, 4, 6	Unique	Last Branch Record 7
					See description of MSR_LASTBRANCH_0 at 680H.
688H	1672	MSR_LASTBRANCH_8_FROM_IP	3, 4, 6	Unique	Last Branch Record 8
					See description of MSR_LASTBRANCH_0 at 680H.
689H	1673	MSR_LASTBRANCH_9_FROM_IP	3, 4, 6	Unique	Last Branch Record 9
					See description of MSR_LASTBRANCH_0 at 680H.
68AH	1674	MSR_LASTBRANCH_10_FROM_IP	3, 4, 6	Unique	Last Branch Record 10
					See description of MSR_LASTBRANCH_0 at 680H.
68BH	1675	MSR_LASTBRANCH_11_FROM_IP	3, 4, 6	Unique	Last Branch Record 11
					See description of MSR_LASTBRANCH_0 at 680H.
68CH	1676	MSR_LASTBRANCH_12_FROM_IP	3, 4, 6	Unique	Last Branch Record 12
					See description of MSR_LASTBRANCH_0 at 680H.

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

	ister ress	Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
68DH	1677	MSR_LASTBRANCH_13_FROM_IP	3, 4, 6	Unique	Last Branch Record 13 See description of MSR_LASTBRANCH_0 at 680H.
68EH	1678	MSR_LASTBRANCH_14_FROM_IP	3, 4, 6	Unique	Last Branch Record 14 See description of MSR_LASTBRANCH_0 at 680H.
68FH	1679	MSR_LASTBRANCH_15_FROM_IP	3, 4, 6	Unique	Last Branch Record 15 See description of MSR_LASTBRANCH_0 at 680H.
6СОН	1728	MSR_LASTBRANCH_O_TO_IP	3, 4, 6	Unique	Last Branch Record 0 (R/W) One of 16 pairs of last branch record registers on the last branch record stack (6C0H-6CFH). This part of the stack contains pointers to the destination instruction for one of the last 16 branches, exceptions, or interrupts that the processor took. See Section 18.12, "Last Branch, Call Stack, Interrupt, and Exception Recording for Processors based on Skylake Microarchitecture."
6C1H	1729	MSR_LASTBRANCH_1_TO_IP	3, 4, 6	Unique	Last Branch Record 1 See description of MSR_LASTBRANCH_0 at 6COH.
6C2H	1730	MSR_LASTBRANCH_2_TO_IP	3, 4, 6	Unique	Last Branch Record 2 See description of MSR_LASTBRANCH_0 at 6C0H.
6C3H	1731	MSR_LASTBRANCH_3_TO_IP	3, 4, 6	Unique	Last Branch Record 3 See description of MSR_LASTBRANCH_0 at 6C0H.
6C4H	1732	MSR_LASTBRANCH_4_TO_IP	3, 4, 6	Unique	Last Branch Record 4 See description of MSR_LASTBRANCH_0 at 6COH.
6C5H	1733	MSR_LASTBRANCH_5_TO_IP	3, 4, 6	Unique	Last Branch Record 5 See description of MSR_LASTBRANCH_0 at 6C0H.
6C6H	1734	MSR_LASTBRANCH_6_TO_IP	3, 4, 6	Unique	Last Branch Record 6 See description of MSR_LASTBRANCH_0 at 6COH.
6C7H	1735	MSR_LASTBRANCH_7_TO_IP	3, 4, 6	Unique	Last Branch Record 7 See description of MSR_LASTBRANCH_0 at 6COH.
6C8H	1736	MSR_LASTBRANCH_8_TO_IP	3, 4, 6	Unique	Last Branch Record 8 See description of MSR_LASTBRANCH_0 at 6COH.

Table 2-55. MSRs in the Pentium® 4 and Intel® Xeon® Processors (Contd.)

Register Address		Register Name Fields and Flags	Model Avail-	Shared/ Unique ¹	Bit Description
Hex	Dec		ability		
6C9H	1737	MSR_LASTBRANCH_9_TO_IP	3, 4, 6	Unique	Last Branch Record 9
					See description of MSR_LASTBRANCH_0 at 6C0H.
6CAH	1738	MSR_LASTBRANCH_10_TO_IP	3, 4, 6	Unique	Last Branch Record 10
					See description of MSR_LASTBRANCH_0 at 6C0H.
6CBH	1739	MSR_LASTBRANCH_11_TO_IP	3, 4, 6	Unique	Last Branch Record 11
					See description of MSR_LASTBRANCH_0 at 6C0H.
6CCH	1740	MSR_LASTBRANCH_12_TO_IP	3, 4, 6	Unique	Last Branch Record 12
					See description of MSR_LASTBRANCH_0 at 6C0H.
6CDH	1741	MSR_LASTBRANCH_13_TO_IP	3, 4, 6	Unique	Last Branch Record 13
					See description of MSR_LASTBRANCH_0 at 6C0H.
6CEH	1742	MSR_LASTBRANCH_14_TO_IP	3, 4, 6	Unique	Last Branch Record 14
					See description of MSR_LASTBRANCH_0 at 6C0H.
6CFH	1743	MSR_LASTBRANCH_15_TO_IP	3, 4, 6	Unique	Last Branch Record 15
					See description of MSR_LASTBRANCH_0 at 6C0H.
C000_		IA32_EFER	3, 4, 6	Unique	Extended Feature Enables
H0800					See Table 2-2.
C000_		IA32_STAR	3, 4, 6	Unique	System Call Target Address (R/W)
0081H					See Table 2-2.
C000_		IA32_LSTAR	3, 4, 6	Unique	IA-32e Mode System Call Target Address (R/W)
0082H					See Table 2-2.
C000_ 0084H		IA32_FMASK	3, 4, 6	Unique	System Call Flag Mask (R/W)
					See Table 2-2.
C000_ 0100H		IA32_FS_BASE	3, 4, 6	Unique	Map of BASE Address of FS (R/W)
					See Table 2-2.
C000_ 0101H		IA32_GS_BASE	3, 4, 6	Unique	Map of BASE Address of GS (R/W)
				1	See Table 2-2.
C000_ 0102H		IA32_KERNEL_GS_BASE	3, 4, 6	Unique	Swap Target of BASE Address of GS (R/W)
010211					See Table 2-2.

NOTES

^{1.} For HT-enabled processors, there may be more than one logical processors per physical unit. If an MSR is Shared, this means that one MSR is shared between logical processors. If an MSR is unique, this means that each logical processor has its own MSR.

2.19.1 MSRs Unique to Intel® Xeon® Processor MP with L3 Cache

The MSRs listed in Table 2-56 apply to $Intel^{\circledR}$ Xeon $^{\circledR}$ Processor MP with up to 8MB level three cache. These processors can be detected by enumerating the deterministic cache parameter leaf of CPUID instruction (with EAX = 4 as input) to detect the presence of the third level cache, and with CPUID reporting family encoding 0FH, model encoding 3 or 4 (see CPUID instruction for more details).

Table 2-56. MSRs Unique to 64-bit Intel® Xeon® Processor MP with Up to an 8 MB L3 Cache

Register Address		Register Name Model Avail- Fields and Flags ability		Shared/ Unique	Bit Description	
107CCH		MSR_IFSB_BUSQ0	3, 4	Shared	IFSB BUSQ Event Control and Counter Register (R/W)	
					See Section 20.6.6, "Performance Monitoring on 64-bit Intel Xeon Processor MP with Up to 8-MByte L3 Cache."	
107CDH		MSR_IFSB_BUSQ1	3, 4	Shared	IFSB BUSQ Event Control and Counter Register (R/W)	
107CEH		MSR_IFSB_SNPQ0	3, 4	Shared	IFSB SNPQ Event Control and Counter Register (R/W)	
					See Section 20.6.6, "Performance Monitoring on 64-bit Intel Xeon Processor MP with Up to 8-MByte L3 Cache."	
107CFH		MSR_IFSB_SNPQ1	3, 4	Shared	IFSB SNPQ Event Control and Counter Register (R/W)	
107D0H		MSR_EFSB_DRDY0	3, 4	Shared	EFSB DRDY Event Control and Counter Register (R/W)	
					See Section 20.6.6, "Performance Monitoring on 64-bit Intel Xeon Processor MP with Up to 8-MByte L3 Cache."	
107D1H		MSR_EFSB_DRDY1	3, 4	Shared	EFSB DRDY Event Control and Counter Register (R/W)	
107D2H		MSR_IFSB_CTL6	3, 4	Shared	IFSB Latency Event Control Register (R/W)	
					See Section 20.6.6, "Performance Monitoring on 64-bit Intel Xeon Processor MP with Up to 8-MByte L3 Cache."	
107D3H		MSR_IFSB_CNTR7	3, 4	Shared	IFSB Latency Event Counter Register (R/W)	
					See Section 20.6.6, "Performance Monitoring on 64-bit Intel Xeon Processor MP with Up to 8-MByte L3 Cache."	

The MSRs listed in Table 2-57 apply to Intel[®] Xeon[®] Processor 7100 series. These processors can be detected by enumerating the deterministic cache parameter leaf of CPUID instruction (with EAX = 4 as input) to detect the presence of the third level cache, and with CPUID reporting family encoding 0FH, model encoding 6 (See CPUID instruction for more details.). The performance monitoring MSRs listed in Table 2-57 are shared between logical processors in the same core, but are replicated for each core.

Table 2-57. MSRs Unique to Intel® Xeon® Processor 7100 Series

Register Address		Register Name Fields and Flags	Model Avail- ability	Shared/ Unique	Bit Description
107CCH		MSR_EMON_L3_CTR_CTL0	6	Shared	GBUSQ Event Control and Counter Register (R/W)
					See Section 20.6.6, "Performance Monitoring on 64-bit Intel Xeon Processor MP with Up to 8-MByte L3 Cache."

Table 2-57. MSRs Unique to Intel® Xeon® Processor 7100 Series (Contd.)

Register Address	Register Name Fields and Flags	Model Avail- ability	Shared/ Unique	Bit Description
107CDH	MSR_EMON_L3_CTR_CTL1	6	Shared	GBUSQ Event Control and Counter Register (R/W)
107CEH	MSR_EMON_L3_CTR_CTL2	6	Shared	GSNPQ Event Control and Counter Register (R/W) See Section 20.6.6, "Performance Monitoring on 64-bit Intel Xeon Processor MP with Up to 8- MByte L3 Cache."
107CFH	MSR_EMON_L3_CTR_CTL3	6	Shared	GSNPQ Event Control and Counter Register (R/W)
107D0H	MSR_EMON_L3_CTR_CTL4	6	Shared	FSB Event Control and Counter Register (R/W) See Section 20.6.6, "Performance Monitoring on 64-bit Intel Xeon Processor MP with Up to 8- MByte L3 Cache."
107D1H	MSR_EMON_L3_CTR_CTL5	6	Shared	FSB Event Control and Counter Register (R/W)
107D2H	MSR_EMON_L3_CTR_CTL6	6	Shared	FSB Event Control and Counter Register (R/W)
107D3H	MSR_EMON_L3_CTR_CTL7	6	Shared	FSB Event Control and Counter Register (R/W)

2.20 MSRS IN INTEL® CORE™ SOLO AND INTEL® CORE™ DUO PROCESSORS

Model-specific registers (MSRs) for Intel Core Solo, Intel Core Duo processors, and Dual-core Intel Xeon processor LV are listed in Table 2-58. The column "Shared/Unique" applies to Intel Core Duo processor. "Unique" means each processor core has a separate MSR, or a bit field in an MSR governs only a core independently. "Shared" means the MSR or the bit field in an MSR address governs the operation of both processor cores.

Table 2-58. MSRs in Intel® Core™ Solo, Intel® Core™ Duo Processors, and Dual-Core Intel® Xeon® Processor LV

_	ister ress	Register Name	Shared/ Unique	Bit Description
Hex	Dec			
OH	0	P5_MC_ADDR	Unique	See Section 2.23, "MSRs in Pentium Processors," and see Table 2-2.
1H	1	P5_MC_TYPE	Unique	See Section 2.23, "MSRs in Pentium Processors," and see Table 2-2.
6H	6	IA32_MONITOR_FILTER_SIZE	Unique	See Section 9.10.5, "Monitor/Mwait Address Range Determination," and see Table 2-2.
10H	16	IA32_TIME_STAMP_COUNTER	Unique	See Section 18.17, "Time-Stamp Counter," and see Table 2-2.
17H	23	IA32_PLATFORM_ID	Shared	Platform ID (R) See Table 2-2. The operating system can use this MSR to determine "slot" information for the processor and the proper microcode update to load.
1BH	27	IA32_APIC_BASE	Unique	See Section 11.4.4, "Local APIC Status and Location," and see Table 2-2.

Table 2-58. MSRs in Intel® Core™ Solo, Intel® Core™ Duo Processors, and Dual-Core Intel® Xeon® Processor LV

Regi Addi	ster	Register Name	Shared/ Unique	Bit Description
Hex	Dec			
2AH	42	MSR_EBL_CR_POWERON	Shared	Processor Hard Power-On Configuration (R/W) Enables and disables processor features; (R) indicates current processor configuration.
		0		Reserved
		1		Data Error Checking Enable (R/W) 1 = Enabled; 0 = Disabled Note: Not all processor implements R/W.
		2		Response Error Checking Enable (R/W) 1 = Enabled; 0 = Disabled Note: Not all processor implements R/W.
		3		MCERR# Drive Enable (R/W) 1 = Enabled; 0 = Disabled Note: Not all processor implements R/W.
		4		Address Parity Enable (R/W) 1 = Enabled; 0 = Disabled Note: Not all processor implements R/W.
		6: 5		Reserved
		7		BINIT# Driver Enable (R/W) 1 = Enabled; 0 = Disabled Note: Not all processor implements R/W.
		8		Output Tri-state Enabled (R/O) 1 = Enabled; O = Disabled
		9		Execute BIST (R/O) 1 = Enabled; 0 = Disabled
		10		MCERR# Observation Enabled (R/O) 1 = Enabled; 0 = Disabled
		11		Reserved
		12		BINIT# Observation Enabled (R/O) 1 = Enabled; O = Disabled
		13		Reserved
		14		1 MByte Power on Reset Vector (R/O) 1 = 1 MByte; 0 = 4 GBytes
		15		Reserved
		17:16		APIC Cluster ID (R/O)
		18		System Bus Frequency (R/O) 0 = 100 MHz 1 = Reserved
		19		Reserved
		21: 20		Symmetric Arbitration ID (R/O)
		26:22		Clock Frequency Ratio (R/O)

Table 2-58. MSRs in Intel® Core™ Solo, Intel® Core™ Duo Processors, and Dual-Core Intel® Xeon® Processor LV

Regi Addı	ster	Register Name	Shared/ Unique	Bit Description
Hex	Dec			
ЗАН	58	IA32_FEATURE_CONTROL	Unique	Control Features in IA-32 Processor (R/W) See Table 2-2.
40H	64	MSR_LASTBRANCH_0	Unique	Last Branch Record 0 (R/W) One of 8 last branch record registers on the last branch record stack: bits 31-0 hold the 'from' address and bits 63-32 hold the 'to' address. See also: Last Branch Record Stack TOS at 1C9H Section 18.15, "Last Branch, Interrupt, and Exception Recording (Pentium M Processors)."
41H	65	MSR_LASTBRANCH_1	Unique	Last Branch Record 1 (R/W) See description of MSR_LASTBRANCH_0.
42H	66	MSR_LASTBRANCH_2	Unique	Last Branch Record 2 (R/W) See description of MSR_LASTBRANCH_0.
43H	67	MSR_LASTBRANCH_3	Unique	Last Branch Record 3 (R/W) See description of MSR_LASTBRANCH_0.
44H	68	MSR_LASTBRANCH_4	Unique	Last Branch Record 4 (R/W) See description of MSR_LASTBRANCH_0.
45H	69	MSR_LASTBRANCH_5	Unique	Last Branch Record 5 (R/W) See description of MSR_LASTBRANCH_0.
46H	70	MSR_LASTBRANCH_6	Unique	Last Branch Record 6 (R/W) See description of MSR_LASTBRANCH_0.
47H	71	MSR_LASTBRANCH_7	Unique	Last Branch Record 7 (R/W) See description of MSR_LASTBRANCH_0.
79H	121	IA32_BIOS_UPDT_TRIG	Unique	BIOS Update Trigger Register (W) See Table 2-2.
8BH	139	IA32_BIOS_SIGN_ID	Unique	BIOS Update Signature ID (R/W) See Table 2-2.
C1H	193	IA32_PMC0	Unique	Performance Counter Register See Table 2-2.
C2H	194	IA32_PMC1	Unique	Performance Counter Register See Table 2-2.
CDH	205	MSR_FSB_FREQ 2:0	Shared	Scaleable Bus Speed (R/O) This field indicates the scaleable bus clock speed: 101B: 100 MHz (FSB 400) 001B: 133 MHz (FSB 533)
		63:3		011B: 167 MHz (FSB 667) 133.33 MHz should be utilized if performing calculation with System Bus Speed when encoding is 101B. 166.67 MHz should be utilized if performing calculation with System Bus Speed when encoding is 001B. Reserved
		00.0		INCOCI VCC

Table 2-58. MSRs in Intel® Core™ Solo, Intel® Core™ Duo Processors, and Dual-Core Intel® Xeon® Processor LV

Regi	ister ress	Register Name	Shared/ Unique	Bit Description
Hex	Dec			
E7H	231	IA32_MPERF	Unique	Maximum Performance Frequency Clock Count (R/W) See Table 2-2.
E8H	232	IA32_APERF	Unique	Actual Performance Frequency Clock Count (R/W) See Table 2-2.
FEH	254	IA32_MTRRCAP	Unique	See Table 2-2.
11EH	281	MSR_BBL_CR_CTL3	Shared	Control Register 3 Used to configure the L2 Cache.
		0		L2 Hardware Enabled (R/O) 1 = If the L2 is hardware-enabled 0 = Indicates if the L2 is hardware-disabled
		7:1		Reserved
		8		L2 Enabled (R/W) 1 = L2 cache has been initialized 0 = Disabled (default) Until this bit is set the processor will not respond to the WBINVD instruction or the assertion of the FLUSH# input.
		22:9		Reserved
		23		L2 Not Present (R/O) 0 = L2 Present 1 = L2 Not Present
		63:24		Reserved
174H	372	IA32_SYSENTER_CS	Unique	See Table 2-2.
175H	373	IA32_SYSENTER_ESP	Unique	See Table 2-2.
176H	374	IA32_SYSENTER_EIP	Unique	See Table 2-2.
179H	377	IA32_MCG_CAP	Unique	See Table 2-2.
17AH	378	IA32_MCG_STATUS	Unique	Global Machine Check Status
		0		RIPV When set, this bit indicates that the instruction addressed by the instruction pointer pushed on the stack (when the machine check was generated) can be used to restart the program. If this bit is cleared, the program cannot be reliably restarted.
		1		EIPV When set, this bit indicates that the instruction addressed by the instruction pointer pushed on the stack (when the machine check was generated) is directly associated with the error.

Table 2-58. MSRs in Intel® Core™ Solo, Intel® Core™ Duo Processors, and Dual-Core Intel® Xeon® Processor LV

Register Address		Register Name	Shared/ Unique	Bit Description	
Hex	Dec				
		2		MCIP When set, this bit indicates that a machine check has been generated. If a second machine check is detected while this bit is still set, the processor enters a shutdown state. Software should write this bit to 0 after processing a machine check exception.	
		63:3		Reserved	
186H	390	IA32_PERFEVTSEL0	Unique	See Table 2-2.	
187H	391	IA32_PERFEVTSEL1	Unique	See Table 2-2.	
198H	408	IA32_PERF_STATUS	Shared	See Table 2-2.	
199H	409	IA32_PERF_CTL	Unique	See Table 2-2.	
19AH	410	IA32_CLOCK_MODULATION	Unique	Clock Modulation (R/W) See Table 2-2.	
19BH	411	IA32_THERM_INTERRUPT	Unique	Thermal Interrupt Control (R/W) See Table 2-2. See Section 15.8.2, "Thermal Monitor."	
19CH	412	IA32_THERM_STATUS	Unique	Thermal Monitor Status (R/W) See Table 2-2. See Section 15.8.2, "Thermal Monitor".	
19DH	413	MSR_THERM2_CTL	Unique	Thermal Monitor 2 Control	
		15:0		Reserved	
		16		TM_SELECT (R/W)	
				Mode of automatic thermal monitor:	
				 0 = Thermal Monitor 1 (thermally-initiated on-die modulation of the stop-clock duty cycle) 1 = Thermal Monitor 2 (thermally-initiated frequency transitions) 	
				If bit 3 of the IA32_MISC_ENABLE register is cleared, TM_SELECT has no effect. Neither TM1 nor TM2 will be enabled.	
		63:16		Reserved	
1AOH	416	IA32_MISC_ENABLE		Enable Miscellaneous Processor Features (R/W) Allows a variety of processor functions to be enabled and disabled.	
		2:0		Reserved	
		3	Unique	Automatic Thermal Control Circuit Enable (R/W) See Table 2-2.	
		6:4		Reserved	
		7	Shared	Performance Monitoring Available (R) See Table 2-2.	

Table 2-58. MSRs in Intel® Core™ Solo, Intel® Core™ Duo Processors, and Dual-Core Intel® Xeon® Processor LV

Regi		Register Name	Shared/ Unique	Bit Description
Hex	Dec			
		9:8		Reserved
		10	Shared	FERR# Multiplexing Enable (R/W) 1 = FERR# asserted by the processor to indicate a pending break event within the processor 0 = Indicates compatible FERR# signaling behavior This bit must be set to 1 to support XAPIC interrupt model usage.
		11	Shared	Branch Trace Storage Unavailable (R/O) See Table 2-2.
		12		Reserved
		13	Shared	TM2 Enable (R/W)
				When this bit is set (1) and the thermal sensor indicates that the die temperature is at the pre-determined threshold, the Thermal Monitor 2 mechanism is engaged. TM2 will reduce the bus to core ratio and voltage according to the value last written to MSR_THERM2_CTL bits 15:0.
				When this bit is clear (0, default), the processor does not change the VID signals or the bus to core ratio when the processor enters a thermal managed state.
				If the TM2 feature flag (ECX[8]) is not set to 1 after executing CPUID with EAX = 1, then this feature is not supported and BIOS must not alter the contents of this bit location. The processor is operating out of spec if both this bit and the TM1 bit are set to disabled states.
		15:14		Reserved
		16	Shared	Enhanced Intel SpeedStep Technology Enable (R/W) 1 = Enhanced Intel SpeedStep Technology enabled
		18	Shared	ENABLE MONITOR FSM (R/W) See Table 2-2.
		19		Reserved
		22	Shared	Limit CPUID Maxval (R/W) See Table 2-2. Setting this bit may cause behavior in software that depends
				on the availability of CPUID leaves greater than 2.
		33:23		Reserved
		34	Shared	XD Bit Disable (R/W)
				See Table 2-2.
		63:35		Reserved
1C9H	457	MSR_LASTBRANCH_TOS	Unique	Last Branch Record Stack TOS (R/W) Contains an index (bits 0-3) that points to the MSR containing the most recent branch record.
				See MSR_LASTBRANCH_0_FROM_IP (at 40H).

Table 2-58. MSRs in Intel® Core™ Solo, Intel® Core™ Duo Processors, and Dual-Core Intel® Xeon® Processor LV

Regi	ister ress	Register Name	Shared/ Unique	Bit Description
Hex	Dec			
1D9H	473	IA32_DEBUGCTL	Unique	Debug Control (R/W)
				Controls how several debug features are used. Bit definitions are discussed in Table 2-2.
1DDH	477	MSR_LER_FROM_LIP	Unique	Last Exception Record From Linear IP (R)
				Contains a pointer to the last branch instruction that the processor executed prior to the last exception that was generated or the last interrupt that was handled.
1DEH	478	MSR_LER_TO_LIP	Unique	Last Exception Record To Linear IP (R)
				This area contains a pointer to the target of the last branch instruction that the processor executed prior to the last exception that was generated or the last interrupt that was handled.
200H	512	MTRRphysBase0	Unique	Memory Type Range Registers
201H	513	MTRRphysMask0	Unique	Memory Type Range Registers
202H	514	MTRRphysBase1	Unique	Memory Type Range Registers
203H	515	MTRRphysMask1	Unique	Memory Type Range Registers
204H	516	MTRRphysBase2	Unique	Memory Type Range Registers
205H	517	MTRRphysMask2	Unique	Memory Type Range Registers
206H	518	MTRRphysBase3	Unique	Memory Type Range Registers
207H	519	MTRRphysMask3	Unique	Memory Type Range Registers
208H	520	MTRRphysBase4	Unique	Memory Type Range Registers
209H	521	MTRRphysMask4	Unique	Memory Type Range Registers
20AH	522	MTRRphysBase5	Unique	Memory Type Range Registers
20BH	523	MTRRphysMask5	Unique	Memory Type Range Registers
20CH	524	MTRRphysBase6	Unique	Memory Type Range Registers
20DH	525	MTRRphysMask6	Unique	Memory Type Range Registers
20EH	526	MTRRphysBase7	Unique	Memory Type Range Registers
20FH	527	MTRRphysMask7	Unique	Memory Type Range Registers
250H	592	MTRRfix64K_00000	Unique	Memory Type Range Registers
258H	600	MTRRfix16K_80000	Unique	Memory Type Range Registers
259H	601	MTRRfix16K_A0000	Unique	Memory Type Range Registers
268H	616	MTRRfix4K_C0000	Unique	Memory Type Range Registers
269H	617	MTRRfix4K_C8000	Unique	Memory Type Range Registers
26AH	618	MTRRfix4K_D0000	Unique	Memory Type Range Registers
26BH	619	MTRRfix4K_D8000	Unique	Memory Type Range Registers
26CH	620	MTRRfix4K_E0000	Unique	Memory Type Range Registers
26DH	621	MTRRfix4K_E8000	Unique	Memory Type Range Registers
26EH	622	MTRRfix4K_F0000	Unique	Memory Type Range Registers
26FH	623	MTRRfix4K_F8000	Unique	Memory Type Range Registers

Table 2-58. MSRs in Intel® Core™ Solo, Intel® Core™ Duo Processors, and Dual-Core Intel® Xeon® Processor LV

Reg	ister ress	Register Name	Shared/ Unique	Bit Description
Hex	Dec			
2FFH	767	IA32_MTRR_DEF_TYPE	Unique	Default Memory Types (R/W) See Table 2-2. See Section 12.11.2.1, "IA32_MTRR_DEF_TYPE MSR."
400H	1024	IA32_MCO_CTL	Unique	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
401H	1025	IA32_MCO_STATUS	Unique	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
402H	1026	IA32_MCO_ADDR	Unique	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs." The IA32_MCO_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MCO_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
404H	1028	IA32_MC1_CTL	Unique	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
405H	1029	IA32_MC1_STATUS	Unique	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
406H	1030	IA32_MC1_ADDR	Unique	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs." The IA32_MC1_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MC1_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
408H	1032	IA32_MC2_CTL	Unique	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
409H	1033	IA32_MC2_STATUS	Unique	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
40AH	1034	IA32_MC2_ADDR	Unique	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs." The IA32_MC2_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MC2_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
40CH	1036	MSR_MC4_CTL	Unique	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
40DH	1037	MSR_MC4_STATUS	Unique	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
40EH	1038	MSR_MC4_ADDR	Unique	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs." The MSR_MC4_ADDR register is either not implemented or contains no address if the ADDRV flag in the MSR_MC4_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
410H	1040	IA32_MC3_CTL		See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
411H	1041	IA32_MC3_STATUS		See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
412H	1042	MSR_MC3_ADDR	Unique	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs." The MSR_MC3_ADDR register is either not implemented or contains no address if the ADDRV flag in the MSR_MC3_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.

Table 2-58. MSRs in Intel® Core™ Solo, Intel® Core™ Duo Processors, and Dual-Core Intel® Xeon® Processor LV

Regi	ister ress	Register Name	Shared/ Unique	Bit Description
Hex	Dec			
413H	1043	MSR_MC3_MISC	Unique	Machine Check Error Reporting Register - contains additional information describing the machine-check error if the MISCV flag in the IA32_MCi_STATUS register is set.
414H	1044	MSR_MC5_CTL	Unique	Machine Check Error Reporting Register - controls signaling of #MC for errors produced by a particular hardware unit (or group of hardware units).
415H	1045	MSR_MC5_STATUS	Unique	Machine Check Error Reporting Register - contains information related to a machine-check error if its VAL (valid) flag is set. Software is responsible for clearing IA32_MCi_STATUS MSRs by explicitly writing 0s to them; writing 1s to them causes a general-protection exception.
416H	1046	MSR_MC5_ADDR	Unique	Machine Check Error Reporting Register - contains the address of the code or data memory location that produced the machine-check error if the ADDRV flag in the IA32_MCi_STATUS register is set.
417H	1047	MSR_MC5_MISC	Unique	Machine Check Error Reporting Register - contains additional information describing the machine-check error if the MISCV flag in the IA32_MCi_STATUS register is set.
480H	1152	IA32_VMX_BASIC	Unique	Reporting Register of Basic VMX Capabilities (R/O) See Table 2-2. See Appendix A.1, "Basic VMX Information". (If CPUID.01H:ECX.[bit 5])
481H	1153	IA32_VMX_PINBASED_CTLS	Unique	Capability Reporting Register of Pin-Based VM-Execution Controls (R/O) See Appendix A.3, "VM-Execution Controls". (If CPUID.01H:ECX.[bit 5])
482H	1154	IA32_VMX_PROCBASED_CTLS	Unique	Capability Reporting Register of Primary Processor-Based VM-Execution Controls (R/O) See Appendix A.3, "VM-Execution Controls". (If CPUID.01H:ECX.[bit 5])
483H	1155	IA32_VMX_EXIT_CTLS	Unique	Capability Reporting Register of VM-Exit Controls (R/O) See Appendix A.4, "VM-Exit Controls". (If CPUID.01H:ECX.[bit 5])
484H	1156	IA32_VMX_ENTRY_CTLS	Unique	Capability Reporting Register of VM-Entry Controls (R/O) See Appendix A.5, "VM-Entry Controls". (If CPUID.01H:ECX.[bit 5])
485H	1157	IA32_VMX_MISC	Unique	Reporting Register of Miscellaneous VMX Capabilities (R/O) See Appendix A.6, "Miscellaneous Data". (If CPUID.01H:ECX.[bit 5])
486H	1158	IA32_VMX_CRO_FIXEDO	Unique	Capability Reporting Register of CRO Bits Fixed to 0 (R/O) See Appendix A.7, "VMX-Fixed Bits in CRO". (If CPUID.01H:ECX.[bit 5])

Table 2-58. MSRs in Intel® Core™ Solo, Intel® Core™ Duo Processors, and Dual-Core Intel® Xeon® Processor LV

Regi Add		Register Name	Shared/ Unique	Bit Description
Hex	Dec			
487H	1159	IA32_VMX_CRO_FIXED1	Unique	Capability Reporting Register of CRO Bits Fixed to 1 (R/O) See Appendix A.7, "VMX-Fixed Bits in CRO". (If CPUID.01H:ECX.[bit 5])
488H	1160	IA32_VMX_CR4_FIXED0	Unique	Capability Reporting Register of CR4 Bits Fixed to 0 (R/0) See Appendix A.8, "VMX-Fixed Bits in CR4". (If CPUID.01H:ECX.[bit 5])
489H	1161	IA32_VMX_CR4_FIXED1	Unique	Capability Reporting Register of CR4 Bits Fixed to 1 (R/O) See Appendix A.8, "VMX-Fixed Bits in CR4". (If CPUID.01H:ECX.[bit 5])
48AH	1162	IA32_VMX_VMCS_ENUM	Unique	Capability Reporting Register of VMCS Field Enumeration (R/0) See Appendix A.9, "VMCS Enumeration". (If CPUID.01H:ECX.[bit 5])
48BH	1163	IA32_VMX_PROCBASED_CTLS2	Unique	Capability Reporting Register of Secondary Processor-Based VM-Execution Controls (R/O) See Appendix A.3, "VM-Execution Controls". (If CPUID.01H:ECX.[bit 5] and IA32_VMX_PROCBASED_CTLS[bit 63])
600H	1536	IA32_DS_AREA	Unique	DS Save Area (R/W) See Table 2-2. See Section 20.6.3.4, "Debug Store (DS) Mechanism."
		31:0		DS Buffer Management Area Linear address of the first byte of the DS buffer management area.
		63:32		Reserved
0080H		IA32_EFER	Unique	See Table 2-2.
		10:0		Reserved
		11		Execute Disable Bit Enable
		63:12		Reserved

2.21 MSRS IN THE PENTIUM M PROCESSOR

Model-specific registers (MSRs) for the Pentium M processor are similar to those described in Section 2.22 for P6 family processors. The following table describes new MSRs and MSRs whose behavior has changed on the Pentium M processor.

Table 2-59. MSRs in Pentium M Processors

Register Address		Register Name / Bit Fields	Bit Description
Hex	Dec		
OH	0	P5_MC_ADDR	See Section 2.23, "MSRs in Pentium Processors."

Table 2-59. MSRs in Pentium M Processors (Contd.)

Regi Addi		Register Name / Bit Fields	Bit Description
Hex	Dec	1	
1H	1	P5_MC_TYPE	See Section 2.23, "MSRs in Pentium Processors."
10H	16	IA32_TIME_STAMP_COUNTER	See Section 18.17, "Time-Stamp Counter," and see Table 2-2.
17H	23	IA32_PLATFORM_ID	Platform ID (R)
			See Table 2-2.
			The operating system can use this MSR to determine "slot" information for the processor and the proper microcode update to load.
2AH	42	MSR_EBL_CR_POWERON	Processor Hard Power-On Configuration
			(R/W) Enables and disables processor features.
			(R) Indicates current processor configuration.
		0	Reserved
		1	Data Error Checking Enable (R)
			0 = Disabled
			Always 0 on the Pentium M processor.
		2	Response Error Checking Enable (R)
			0 = Disabled
			Always 0 on the Pentium M processor.
		3	MCERR# Drive Enable (R)
			0 = Disabled
			Always 0 on the Pentium M processor.
		4	Address Parity Enable (R)
			0 = Disabled
			Always 0 on the Pentium M processor.
		6:5	Reserved
		7	BINIT# Driver Enable (R)
			1 = Enabled; 0 = Disabled
			Always 0 on the Pentium M processor.
		8	Output Tri-state Enabled (R/O)
			1 = Enabled; 0 = Disabled
		9	Execute BIST (R/O)
			1 = Enabled; 0 = Disabled
		10	MCERR# Observation Enabled (R/O)
			1 = Enabled; 0 = Disabled
			Always 0 on the Pentium M processor.
		11	Reserved
		12	BINIT# Observation Enabled (R/O)
			1 = Enabled; 0 = Disabled
			Always 0 on the Pentium M processor.
		13	Reserved

Table 2-59. MSRs in Pentium M Processors (Contd.)

Regi Add		Register Name / Bit Fields	Bit Description
Hex	Dec		
		14	1 MByte Power on Reset Vector (R/O) 1 = 1 MByte; 0 = 4 GBytes Always 0 on the Pentium M processor.
		15	Reserved
		17:16	APIC Cluster ID (R/O)
			Always 00B on the Pentium M processor.
		18	System Bus Frequency (R/O) 0 = 100 MHz 1 = Reserved Always 0 on the Pentium M processor.
		19	Reserved
		21: 20	Symmetric Arbitration ID (R/O)
			Always 00B on the Pentium M processor.
		26:22	Clock Frequency Ratio (R/O)
40H	64	MSR_LASTBRANCH_0	Last Branch Record 0 (R/W)
			One of 8 last branch record registers on the last branch record stack: bits 31-0 hold the 'from' address and bits 63-32 hold the to address.
			 See also: Last Branch Record Stack TOS at 1C9H. Section 18.15, "Last Branch, Interrupt, and Exception Recording (Pentium M Processors)".
41H	65	MSR_LASTBRANCH_1	Last Branch Record 1 (R/W)
			See description of MSR_LASTBRANCH_0.
42H	66	MSR_LASTBRANCH_2	Last Branch Record 2 (R/W)
			See description of MSR_LASTBRANCH_0.
43H	67	MSR_LASTBRANCH_3	Last Branch Record 3 (R/W)
			See description of MSR_LASTBRANCH_0.
44H	68	MSR_LASTBRANCH_4	Last Branch Record 4 (R/W)
			See description of MSR_LASTBRANCH_0.
45H	69	MSR_LASTBRANCH_5	Last Branch Record 5 (R/W)
			See description of MSR_LASTBRANCH_0.
46H	70	MSR_LASTBRANCH_6	Last Branch Record 6 (R/W)
			See description of MSR_LASTBRANCH_0.
47H	71	MSR_LASTBRANCH_7	Last Branch Record 7 (R/W)
			See description of MSR_LASTBRANCH_0.
119H	281	MSR_BBL_CR_CTL	Control Register Used to program L2 commands to be issued via cache configuration accesses mechanism. Also receives L2 lookup response.
		63:0	Reserved
11EH	281	MSR_BBL_CR_CTL3	Control register 3
11011	201	TISK_DDC_CK_CTCJ	Used to configure the L2 Cache.

Table 2-59. MSRs in Pentium M Processors (Contd.)

	ister ress	Register Name / Bit Fields	Bit Description
Hex	Dec	Register Name / Bit Fields	
		0	L2 Hardware Enabled (R/O) 1 = If the L2 is hardware-enabled. 0 = Indicates if the L2 is hardware-disabled.
		4:1	Reserved
		5	ECC Check Enable (R/O)
			This bit enables ECC checking on the cache data bus. ECC is always generated on write cycles.
			0 = Disabled (default) 1 = Enabled
			For the Pentium M processor, ECC checking on the cache data bus is always enabled.
		7:6	Reserved
		8	L2 Enabled (R/W) 1 = L2 cache has been initialized 0 = Disabled (default) Until this bit is set the processor will not respond to the WBINVD instruction or the assertion of the FLUSH# input.
		22:9	Reserved
		23	L2 Not Present (R/O) 0 = L2 Present 1 = L2 Not Present
		63:24	Reserved
179H	377	IA32_MCG_CAP	Read-only register that provides information about the machine-check architecture of the processor.
		7:0	Count (R/O) Indicates the number of hardware unit error reporting banks available in the processor.
		8	IA32_MCG_CTL Present (R/O)
			1 = Indicates that the processor implements the MSR_MCG_CTL register found at MSR 17BH.
			0 = Not supported.
		63:9	Reserved
17AH	378	IA32_MCG_STATUS	Global Machine Check Status
		0	RIPV When set, this bit indicates that the instruction addressed by the instruction pointer pushed on the stack (when the machine check was generated) can be used to restart the program. If this bit is cleared, the program cannot be reliably restarted.
		1	EIPV When set, this bit indicates that the instruction addressed by the instruction pointer pushed on the stack (when the machine check was generated) is directly associated with the error.

Table 2-59. MSRs in Pentium M Processors (Contd.)

Register Address		Register Name / Bit Fields	Bit Description
Hex	Dec		
		2	MCIP
			When set, this bit indicates that a machine check has been generated. If a second machine check is detected while this bit is still set, the processor enters a shutdown state. Software should write this bit to 0 after processing a machine check exception.
		63:3	Reserved
198H	408	IA32_PERF_STATUS	See Table 2-2.
199H	409	IA32_PERF_CTL	See Table 2-2.
19AH	410	IA32_CLOCK_MODULATION	Clock Modulation (R/W).
			See Table 2-2.
			See Section 15.8.3, "Software Controlled Clock Modulation."
19BH	411	IA32_THERM_INTERRUPT	Thermal Interrupt Control (R/W)
			See Table 2-2.
			See Section 15.8.2, "Thermal Monitor."
19CH	412	IA32_THERM_STATUS	Thermal Monitor Status (R/W)
			See Table 2-2.
			See Section 15.8.2, "Thermal Monitor."
19DH	413	MSR_THERM2_CTL	Thermal Monitor 2 Control
		15:0	Reserved
		16	TM_SELECT (R/W)
			Mode of automatic thermal monitor:
			0 = Thermal Monitor 1 (thermally-initiated on-die modulation of the stop-clock duty cycle)
			1 = Thermal Monitor 2 (thermally-initiated frequency transitions)
			If bit 3 of the IA32_MISC_ENABLE register is cleared, TM_SELECT has no effect. Neither TM1 nor TM2 will be enabled.
		63:16	Reserved
1A0H	416	IA32_MISC_ENABLE	Enable Miscellaneous Processor Features (R/W)
			Allows a variety of processor functions to be enabled and disabled.
		2:0	Reserved

Table 2-59. MSRs in Pentium M Processors (Contd.)

Regi Add		Register Name / Bit Fields	Bit Description	
Hex	Dec	-		
		3	Automatic Thermal Control Circuit Enable (R/W) 1 = Setting this bit enables the thermal control circuit (TCC) portion of	
			the Intel Thermal Monitor feature. This allows processor clocks to be automatically modulated based on the processor's thermal sensor operation.	
			0 = Disabled (default).	
			The automatic thermal control circuit enable bit determines if the thermal control circuit (TCC) will be activated when the processor's internal thermal sensor determines the processor is about to exceed its maximum operating temperature.	
			When the TCC is activated and TM1 is enabled, the processors clocks will be forced to a 50% duty cycle. BIOS must enable this feature.	
			The bit should not be confused with the on-demand thermal control circuit enable bit.	
		6:4	Reserved	
		7	Performance Monitoring Available (R)	
			1 = Performance monitoring enabled.0 = Performance monitoring disabled.	
		9:8	Reserved	
		10	FERR# Multiplexing Enable (R/W) 1 = FERR# asserted by the processor to indicate a pending break event within the processor. 0 = Indicates compatible FERR# signaling behavior. This bit must be set to 1 to support XAPIC interrupt model usage.	
			Branch Trace Storage Unavailable (R/O) 1 = Processor doesn't support branch trace storage (BTS) 0 = BTS is supported	
		12	Processor Event Based Sampling Unavailable (R/O) 1 = Processor does not support processor event based sampling (PEBS); 0 = PEBS is supported. The Pentium M processor does not support PEBS.	
		15:13	Reserved	
		16	Enhanced Intel SpeedStep Technology Enable (R/W) 1 = Enhanced Intel SpeedStep Technology enabled. On the Pentium M processor, this bit may be configured to be read-only.	
		22:17	Reserved	
		23	xTPR Message Disable (R/W)	
			When set to 1, xTPR messages are disabled. xTPR messages are optional messages that allow the processor to inform the chipset of its priority. The default is processor specific.	
		63:24	Reserved	

Table 2-59. MSRs in Pentium M Processors (Contd.)

	ister ress	Register Name / Bit Fields	Bit Description
Hex	Dec		
1C9H	457	MSR_LASTBRANCH_TOS	Last Branch Record Stack TOS (R/W)
			Contains an index (bits 0-3) that points to the MSR containing the most recent branch record. See also:
			 MSR_LASTBRANCH_0_FROM_IP (at 40H). Section 18.15, "Last Branch, Interrupt, and Exception Recording (Pentium M Processors)".
1D9H	473	MSR_DEBUGCTLB	Debug Control (R/W)
			Controls how several debug features are used. Bit definitions are discussed in the referenced section.
			See Section 18.15, "Last Branch, Interrupt, and Exception Recording (Pentium M Processors)."
1DDH	477	MSR_LER_TO_LIP	Last Exception Record To Linear IP (R)
			This area contains a pointer to the target of the last branch instruction that the processor executed prior to the last exception that was generated or the last interrupt that was handled.
			See Section 18.15, "Last Branch, Interrupt, and Exception Recording (Pentium M Processors)" and Section 18.16.2, "Last Branch and Last Exception MSRs."
1DEH	478	MSR_LER_FROM_LIP	Last Exception Record From Linear IP (R)
			Contains a pointer to the last branch instruction that the processor executed prior to the last exception that was generated or the last interrupt that was handled.
			See Section 18.15, "Last Branch, Interrupt, and Exception Recording (Pentium M Processors)" and Section 18.16.2, "Last Branch and Last Exception MSRs."
2FFH	767	IA32_MTRR_DEF_TYPE	Default Memory Types (R/W)
			Sets the memory type for the regions of physical memory that are not mapped by the MTRRs.
			See Section 12.11.2.1, "IA32_MTRR_DEF_TYPE MSR."
400H	1024	IA32_MC0_CTL	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
401H	1025	IA32_MC0_STATUS	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
402H	1026	IA32_MC0_ADDR	See Section 14.3.2.3., "IA32_MCi_ADDR MSRs".
			The IA32_MCO_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MCO_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
404H	1028	IA32_MC1_CTL	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
405H	1029	IA32_MC1_STATUS	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
406H	1030	IA32_MC1_ADDR	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
			The IA32_MC1_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MC1_STATUS register is clear.
			When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
408H	1032	IA32_MC2_CTL	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
409H	1033	IA32_MC2_STATUS	See Chapter 16.3.2.2, "IA32_MCi_STATUS MSRS."

Table 2-59. MSRs in Pentium M Processors (Contd.)

	ister ress	Register Name / Bit Fields	Bit Description
Hex	Dec		
40AH	1034	IA32_MC2_ADDR	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs." The IA32_MC2_ADDR register is either not implemented or contains no address if the ADDRV flag in the IA32_MC2_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
40CH	1036	MSR_MC4_CTL	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
40DH	1037	MSR_MC4_STATUS	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
40EH	1038	MSR_MC4_ADDR	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
			The MSR_MC4_ADDR register is either not implemented or contains no address if the ADDRV flag in the MSR_MC4_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
410H	1040	MSR_MC3_CTL	See Section 16.3.2.1, "IA32_MCi_CTL MSRs."
411H	1041	MSR_MC3_STATUS	See Section 16.3.2.2, "IA32_MCi_STATUS MSRS."
412H	1042	MSR_MC3_ADDR	See Section 16.3.2.3, "IA32_MCi_ADDR MSRs."
			The MSR_MC3_ADDR register is either not implemented or contains no address if the ADDRV flag in the MSR_MC3_STATUS register is clear. When not implemented in the processor, all reads and writes to this MSR will cause a general-protection exception.
600H	1536	IA32_DS_AREA	DS Save Area (R/W)
			See Table 2-2.
			Points to the DS buffer management area, which is used to manage the BTS and PEBS buffers. See Section 20.6.3.4, "Debug Store (DS) Mechanism."
		31:0	DS Buffer Management Area
			Linear address of the first byte of the DS buffer management area.
		63:32	Reserved

2.22 MSRS IN THE P6 FAMILY PROCESSORS

The following MSRs are defined for the P6 family processors. The MSRs in this table that are shaded are available only in the Pentium II and Pentium III processors. Beginning with the Pentium 4 processor, some of the MSRs in this list have been designated as "architectural" and have had their names changed. See Table 2-2 for a list of the architectural MSRs.

Table 2-60. MSRs in the P6 Family Processors

Register Address		Register Name / Bit Fields	Bit Description
Hex	Dec		
ОН	0	P5_MC_ADDR	See Section 2.23, "MSRs in Pentium Processors."
1H	1	P5_MC_TYPE	See Section 2.23, "MSRs in Pentium Processors."
10H	16	TSC	See Section 18.17, "Time-Stamp Counter."

Table 2-60. MSRs in the P6 Family Processors (Contd.)

Register Address		Register Name / Bit Fields	Bit Description
Hex	Dec	1	
17H	23	IA32_PLATFORM_ID	Platform ID (R) The operating system can use this MSR to determine "slot" information for the processor and the proper microcode update to load.
		49:0	Reserved
		52:50	Platform Id (R) Contains information concerning the intended platform for the processor. 52 51 50 0 0 0 Processor Flag 0 0 0 1 Processor Flag 1 0 1 0 Processor Flag 2 0 1 1 Processor Flag 3 1 0 0 Processor Flag 4 1 0 1 Processor Flag 5 1 1 0 Processor Flag 6 1 1 1 Processor Flag 7
		56:53	L2 Cache Latency Read.
		59:57	Reserved
		60	Clock Frequency Ratio Read.
		63:61	Reserved
1BH	27	APIC_BASE	Section 11.4.4, "Local APIC Status and Location."
		7:0	Reserved
		8	Boot Strap Processor Indicator Bit 1 = BSP
		10:9	Reserved
		11	APIC Global Enable Bit - Permanent till reset 1 = Enabled 0 = Disabled
		31:12	APIC Base Address.
		63:32	Reserved
2AH	42	EBL_CR_POWERON	Processor Hard Power-On Configuration (R/W) Enables and disables processor features; (R) indicates current processor configuration.
		0	Reserved ¹
		1	Data Error Checking Enable (R/W) 1 = Enabled 0 = Disabled
		2	Response Error Checking Enable FRCERR Observation Enable (R/W) 1 = Enabled 0 = Disabled
		3	AERR# Drive Enable (R/W) 1 = Enabled 0 = Disabled

Table 2-60. MSRs in the P6 Family Processors (Contd.)

Register Address		Register Name / Bit Fields	Bit Description
Hex	Dec		
		4	BERR# Enable for Initiator Bus Requests (R/W) 1 = Enabled 0 = Disabled
		5	Reserved
		6	BERR# Driver Enable for Initiator Internal Errors (R/W) 1 = Enabled 0 = Disabled
		7	BINIT# Driver Enable (R/W) 1 = Enabled 0 = Disabled
		8	Output Tri-state Enabled (R) 1 = Enabled 0 = Disabled
		9	Execute BIST (R) 1 = Enabled 0 = Disabled
		10	AERR# Observation Enabled (R) 1 = Enabled 0 = Disabled
		11	Reserved
		12	BINIT# Observation Enabled (R) 1 = Enabled 0 = Disabled
		13	In Order Queue Depth (R) 1 = 1 0 = 8
		14	1-MByte Power on Reset Vector (R) 1 = 1MByte 0 = 4GBytes
		15	FRC Mode Enable (R) 1 = Enabled 0 = Disabled
		17:16	APIC Cluster ID (R)
		19:18	System Bus Frequency (R) 00 = 66MHz 10 = 100Mhz 01 = 133MHz 11 = Reserved
		21: 20	Symmetric Arbitration ID (R)
		25:22	Clock Frequency Ratio (R)
		26	Low Power Mode Enable (R/W)

Table 2-60. MSRs in the P6 Family Processors (Contd.)

Hex 33H	Dec 51	27 63:28	Clock Frequency Ratio Reserved ¹
33H	51	63:28	<u> </u>
33H	51		D 11
33H	51		Reserved'
		MSR_TEST_CTRL	Test Control Register
		29:0	Reserved
		30	Streaming Buffer Disable
		31	Disable LOCK#
			Assertion for split locked access.
79H	121	BIOS_UPDT_TRIG	BIOS Update Trigger Register.
88H	136	BBL_CR_D0[63:0]	Chunk 0 data register D[63:0]: used to write to and read from the L2
89H	137	BBL_CR_D1[63:0]	Chunk 1 data register D[63:0]: used to write to and read from the L2
8AH	138	BBL_CR_D2[63:0]	Chunk 2 data register D[63:0]: used to write to and read from the L2
8BH	139	BIOS_SIGN/BBL_CR_D3[63:0]	BIOS Update Signature Register or Chunk 3 data register D[63:0]
			Used to write to and read from the L2 depending on the usage model.
C1H	193	PerfCtr0 (PERFCTR0)	Performance Counter Register
			See Table 2-2.
C2H	194	PerfCtr1 (PERFCTR1)	Performance Counter Register
			See Table 2-2.
FEH	254	MTRRcap	Memory Type Range Registers
116H	278	BBL_CR_ADDR [63:0]	Address register: used to send specified address (A31-A3) to L2 during cache initialization accesses.
		BBL_CR_ADDR [63:32]	Reserved,
		BBL_CR_ADDR [31:3]	Address bits [35:3]
		BBL_CR_ADDR [2:0]	Reserved Set to 0.
118H	280	BBL_CR_DECC[63:0]	Data ECC register D[7:0]: used to write ECC and read ECC to/from L2
119H	281	BBL_CR_CTL	Control register: used to program L2 commands to be issued via cache configuration accesses mechanism. Also receives L2 lookup response
		DI CD CTI (C2 22)	Reserved
		BL_CR_CTL[63:22]	Processor number ²
		BBL_CR_CTL[21]	Disable = 1 Enable = 0
			Reserved

Table 2-60. MSRs in the P6 Family Processors (Contd.)

Register Address		Register Name / Bit Fields	Bit Description
Hex	Dec		
		BBL_CR_CTL[20:19] BBL_CR_CTL[18]	User supplied ECC Reserved
		BBL_CR_CTL[17]	L2 Hit
		BBL_CR_CTL[16]	Reserved
		BBL_CR_CTL[15:14]	State from L2
		BBL_CR_CTL[13:12]	Modified - 11,Exclusive - 10, Shared - 01, Invalid - 00
			Way from L2
		BBL_CR_CTL[11:10]	Way 0 - 00, Way 1 - 01, Way 2 - 10, Way 3 - 11
		BBL_CR_CTL[9:8]	Way to L2
		BBL_CR_CTL[7]	Reserved
		BBL_CR_CTL[6:5]	State to L2
		BBL_CR_CTL[4:0]	L2 Command
		01100 01110 01111 00010 00011 010 + MESI encode 111 + MESI encode 100 + MESI encode	Data Read w/ LRU update (RLU) Tag Read w/ Data Read (TRR) Tag Inquire (TI) L2 Control Register Read (CR) L2 Control Register Write (CW) Tag Write w/ Data Read (TWR) Tag Write w/ Data Write (TWW) Tag Write (TW)
11AH	282	BBL_CR_TRIG	Trigger register: used to initiate a cache configuration accesses access, Write only with Data = 0.
11BH	283	BBL_CR_BUSY	Busy register: indicates when a cache configuration accesses L2 command is in progress. D[0] = 1 = BUSY
11EH	286	BBL_CR_CTL3	Control register 3: used to configure the L2 Cache
		BBL_CR_CTL3[63:26]	Reserved
		BBL_CR_CTL3[25]	Cache bus fraction (read only)
		BBL_CR_CTL3[24]	Reserved
		BBL_CR_CTL3[23]	L2 Hardware Disable (read only)
		BBL_CR_CTL3[22:20]	L2 Physical Address Range support
		111 110 101 100 011 010 001 000	64GBytes 32GBytes 16GBytes 8GBytes 4GBytes 2GBytes 1GBytes 1GBytes
		BBL_CR_CTL3[19]	Reserved
		BBL_CR_CTL3[18]	Cache State error checking enable (read/write)

Table 2-60. MSRs in the P6 Family Processors (Contd.)

Register Address		Register Name / Bit Fields	Bit Description	
Hex	Dec			
		BBL_CR_CTL3[17:13 00001 00010 00100 01000 10000	Cache size per bank (read/write) 256KBytes 512KBytes 1MByte 2MByte 4MBytes	
		BBL_CR_CTL3[12:11] BBL_CR_CTL3[10:9] 00 01 10 11	Number of L2 banks (read only) L2 Associativity (read only) Direct Mapped 2 Way 4 Way Reserved	
		BBL_CR_CTL3[8] BBL_CR_CTL3[7] BBL_CR_CTL3[6] BBL_CR_CTL3[5] BBL_CR_CTL3[4:1] BBL_CR_CTL3[0]	L2 Enabled (read/write) CRTN Parity Check Enable (read/write) Address Parity Check Enable (read/write) ECC Check Enable (read/write) L2 Cache Latency (read/write) L2 Configured (read/write)	
174H	372	SYSENTER_CS_MSR	CS register target for CPL 0 code	
175H	373	SYSENTER_ESP_MSR	Stack pointer for CPL 0 stack	
176H	374	SYSENTER_EIP_MSR	CPL 0 code entry point	
179H	377	MCG_CAP	Machine Check Global Control Register	
17AH	378	MCG_STATUS	Machine Check Error Reporting Register - contains information related to a machine-check error if its VAL (valid) flag is set. Software is responsible for clearing IA32_MCi_STATUS MSRs by explicitly writing 0s to them; writing 1s to them causes a general-protection exception.	
17BH	379	MCG_CTL	Machine Check Error Reporting Register - controls signaling of #MC for errors produced by a particular hardware unit (or group of hardware units).	
186H	390	PerfEvtSel0 (EVNTSEL0)	Performance Event Select Register 0 (R/W)	
		7:0	Event Select Refer to Performance Counter section for a list of event encodings.	
		15:8	UMASK (Unit Mask) Unit mask register set to 0 to enable all count options.	
		16	USER Controls the counting of events at Privilege levels of 1, 2, and 3.	
		17	OS Controls the counting of events at Privilege level of 0.	

Table 2-60. MSRs in the P6 Family Processors (Contd.)

Register Address		Register Name / Bit Fields	Bit Description	
Hex	Dec			
		18	E Occurrence/Duration Mode Select 1 = Occurrence 0 = Duration	
		19	PC Enabled the signaling of performance counter overflow via BPO pin	
		20	INT Enables the signaling of counter overflow via input to APIC 1 = Enable 0 = Disable	
		22	ENABLE Enables the counting of performance events in both counters 1 = Enable 0 = Disable	
		23	INV Inverts the result of the CMASK condition 1 = Inverted 0 = Non-Inverted	
		31:24	CMASK (Counter Mask)	
187H	391	PerfEvtSel1 (EVNTSEL1)	Performance Event Select for Counter 1 (R/W)	
		7:0	Event Select Refer to Performance Counter section for a list of event encodings.	
		15:8	UMASK (Unit Mask) Unit mask register set to 0 to enable all count options.	
		16	USER Controls the counting of events at Privilege levels of 1, 2, and 3.	
		17	OS Controls the counting of events at Privilege level of 0.	
		18	E Occurrence/Duration Mode Select. 1 = Occurrence 0 = Duration	
		19	PC Enabled the signaling of performance counter overflow via BPO pin.	

Table 2-60. MSRs in the P6 Family Processors (Contd.)

Register Address		Register Name / Bit Fields	Bit Description	
Hex	Dec			
		20	INT Enables the signaling of counter overflow via input to APIC. 1 = Enable 0 = Disable	
		23	INV Inverts the result of the CMASK condition. 1 = Inverted 0 = Non-Inverted	
		31:24	CMASK (Counter Mask)	
1D9H	473	DEBUGCTLMSR	Enables last branch, interrupt, and exception recording; taken branch breakpoints; the breakpoint reporting pins; and trace messages. This register can be written to using the WRMSR instruction, when operating at privilege level 0 or when in real-address mode.	
		0	Enable/Disable Last Branch Records	
		1	Branch Trap Flag	
		2	Performance Monitoring/Break Point Pins	
		3	Performance Monitoring/Break Point Pins	
		4	Performance Monitoring/Break Point Pins	
		5	Performance Monitoring/Break Point Pins	
		6	Enable/Disable Execution Trace Messages	
		31:7	Reserved	
1DBH	475	LASTBRANCHFROMIP	32-bit register for recording the instruction pointers for the last branch, interrupt, or exception that the processor took prior to a debug exception being generated.	
1DCH	476	LASTBRANCHTOIP	32-bit register for recording the instruction pointers for the last branch, interrupt, or exception that the processor took prior to a debug exception being generated.	
1DDH	477	LASTINTFROMIP	Last INT from IP	
1DEH	478	LASTINTTOIP	Last INT to IP	
200H	512	MTRRphysBase0	Memory Type Range Registers	
201H	513	MTRRphysMask0	Memory Type Range Registers	
202H	514	MTRRphysBase1	Memory Type Range Registers	
203H	515	MTRRphysMask1	Memory Type Range Registers	
204H	516	MTRRphysBase2	Memory Type Range Registers	
205H	517	MTRRphysMask2	Memory Type Range Registers	
206H	518	MTRRphysBase3	Memory Type Range Registers	
207H	519	MTRRphysMask3	Memory Type Range Registers	
208H	520	MTRRphysBase4	Memory Type Range Registers	
209H	521	MTRRphysMask4	Memory Type Range Registers	
20AH	522	MTRRphysBase5	Memory Type Range Registers	

Table 2-60. MSRs in the P6 Family Processors (Contd.)

Register Address		Register Name / Bit Fields	Bit Description	
Hex	Dec			
20BH	523	MTRRphysMask5	Memory Type Range Registers	
20CH	524	MTRRphysBase6	Memory Type Range Registers	
20DH	525	MTRRphysMask6	Memory Type Range Registers	
20EH	526	MTRRphysBase7	Memory Type Range Registers	
20FH	527	MTRRphysMask7	Memory Type Range Registers	
250H	592	MTRRfix64K_00000	Memory Type Range Registers	
258H	600	MTRRfix16K_80000	Memory Type Range Registers	
259H	601	MTRRfix16K_A0000	Memory Type Range Registers	
268H	616	MTRRfix4K_C0000	Memory Type Range Registers	
269H	617	MTRRfix4K_C8000	Memory Type Range Registers	
26AH	618	MTRRfix4K_D0000	Memory Type Range Registers	
26BH	619	MTRRfix4K_D8000	Memory Type Range Registers	
26CH	620	MTRRfix4K_E0000	Memory Type Range Registers	
26DH	621	MTRRfix4K_E8000	Memory Type Range Registers	
26EH	622	MTRRfix4K_F0000	Memory Type Range Registers	
26FH	623	MTRRfix4K_F8000	Memory Type Range Registers	
2FFH	767	MTRRdefType	Memory Type Range Registers	
		2:0	Default memory type	
		10	Fixed MTRR enable	
		11	MTRR Enable	
400H	1024	MCO_CTL	Machine Check Error Reporting Register - controls signaling of #MC for errors produced by a particular hardware unit (or group of hardware units).	
401H	1025	MCO_STATUS	Machine Check Error Reporting Register - contains information related to a machine-check error if its VAL (valid) flag is set. Software is responsible for clearing IA32_MCi_STATUS MSRs by explicitly writing 0s to them; writing 1s to them causes a general-protection exception.	
		15:0	MC_STATUS_MCACOD	
		31:16	MC_STATUS_MSCOD	
		57	MC_STATUS_DAM	
		58	MC_STATUS_ADDRV	
		59	MC_STATUS_MISCV	
		60	MC_STATUS_EN. (Note: For MCO_STATUS only, this bit is hardcoded to 1.)	
		61	MC_STATUS_UC	
		62	MC_STATUS_0	
		63	MC_STATUS_V	
402H	1026	MCO_ADDR		
403H	1027	MCO_MISC	Defined in MCA architecture but not implemented in the P6 family processors.	

Table 2-60. MSRs in the P6 Family Processors (Contd.)

Register Address		Register Name / Bit Fields	Bit Description	
Hex	Dec			
404H	1028	MC1_CTL		
405H	1029	MC1_STATUS	Bit definitions same as MCO_STATUS.	
406H	1030	MC1_ADDR		
407H	1031	MC1_MISC	Defined in MCA architecture but not implemented in the P6 family processors.	
408H	1032	MC2_CTL		
409H	1033	MC2_STATUS	Bit definitions same as MCO_STATUS.	
40AH	1034	MC2_ADDR		
40BH	1035	MC2_MISC	Defined in MCA architecture but not implemented in the P6 family processors.	
40CH	1036	MC4_CTL		
40DH	1037	MC4_STATUS	Bit definitions same as MCO_STATUS, except bits 0, 4, 57, and 61 are hardcoded to 1.	
40EH	1038	MC4_ADDR	Defined in MCA architecture but not implemented in P6 Family processors.	
40FH	1039	MC4_MISC	Defined in MCA architecture but not implemented in the P6 family processors.	
410H	1040	MC3_CTL		
411H	1041	MC3_STATUS	Bit definitions same as MCO_STATUS.	
412H	1042	MC3_ADDR		
413H	1043	MC3_MISC	Defined in MCA architecture but not implemented in the P6 family processors.	

NOTES

1.Bit 0 of this register has been redefined several times, and is no longer used in P6 family processors.

2.The processor number feature may be disabled by setting bit 21 of the BBL_CR_CTL MSR (model-specific register address 119h) to "1". Once set, bit 21 of the BBL_CR_CTL may not be cleared. This bit is write-once. The processor number feature will be disabled until the processor is reset.

3.The Pentium III processor will prevent FSB frequency overclocking with a new shutdown mechanism. If the FSB frequency selected is greater than the internal FSB frequency the processor will shutdown. If the FSB selected is less than the internal FSB frequency the BIOS may choose to use bit 11 to implement its own shutdown policy.

2.23 MSRS IN PENTIUM PROCESSORS

The following MSRs are defined for the Pentium processors. The P5_MC_ADDR, P5_MC_TYPE, and TSC MSRs (named IA32_P5_MC_ADDR, IA32_P5_MC_TYPE, and IA32_TIME_STAMP_COUNTER in the Pentium 4 processor) are architectural; that is, code that accesses these registers will run on Pentium 4 and P6 family processors without generating exceptions (see Section 2.1, "Architectural MSRs"). The CESR, CTR0, and CTR1 MSRs are unique to Pentium processors; code that accesses these registers will generate exceptions on Pentium 4 and P6 family processors.

Table 2-61. MSRs in the Pentium Processor

Regi Addı			
Hex	Dec	Register Name	Bit Description
OH	0	P5_MC_ADDR	See Section 16.10.2, "Pentium Processor Machine-Check Exception Handling."
1H	1	P5_MC_TYPE	See Section 16.10.2, "Pentium Processor Machine-Check Exception Handling."
10H	16	TSC	See Section 18.17, "Time-Stamp Counter."
11H	17	CESR	See Section 20.6.9.1, "Control and Event Select Register (CESR)."
12H	18	CTR0	Section 20.6.9.3, "Events Counted."
13H	19	CTR1	Section 20.6.9.3, "Events Counted."

2.24 MSR INDEX

MSRs of recent processors are indexed here for convenience. IA32 MSRs are excluded from this index.

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_ALF_ESCR0	
0FH	See Table 2-55
MSR_ALF_ESCR1	
0FH	See Table 2-55
MSR_ANY_CORE_CO	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_ANY_GFXE_CO	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_B0_PMON_BOX_CTRL	
06_2EH	See Table 2-17
MSR_B0_PMON_BOX_OVF_CTRL	
06_2EH	See Table 2-17
MSR_B0_PMON_BOX_STATUS	
06_2EH	See Table 2-17
MSR_B0_PMON_CTR0	
06_2EH	See Table 2-17
MSR_B0_PMON_CTR1	
06_2EH	See Table 2-17
MSR_B0_PMON_CTR2	
06_2EH	See Table 2-17
MSR_B0_PMON_CTR3	
06_2EH	See Table 2-17
MSR_B0_PMON_EVNT_SEL0	
06_2EH	See Table 2-17

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_B0_PMON_EVNT_SEL1	
06_2EH	See Table 2-17
MSR_B0_PMON_EVNT_SEL2	
06_2EH	See Table 2-17
MSR_B0_PMON_EVNT_SEL3	
06_2EH	See Table 2-17
MSR_B0_PMON_MASK	
06_2EH	See Table 2-17
MSR_BO_PMON_MATCH	
06_2EH	See Table 2-17
MSR_B1_PMON_BOX_CTRL	
06_2EH	See Table 2-17
MSR_B1_PMON_BOX_OVF_CTRL	
06_2EH	See Table 2-17
MSR_B1_PMON_BOX_STATUS	
06_2EH	See Table 2-17
MSR_B1_PMON_CTR0	
06_2EH	See Table 2-17
MSR_B1_PMON_CTR1	
06_2EH	See Table 2-17
MSR_B1_PMON_CTR2	
06_2EH	See Table 2-17
MSR_B1_PMON_CTR3	
06_2EH	See Table 2-17
MSR_B1_PMON_EVNT_SEL0	
06_2EH	See Table 2-17
MSR_B1_PMON_EVNT_SEL1	
06_2EH	See Table 2-17
MSR_B1_PMON_EVNT_SEL2	
06_2EH	See Table 2-17
MSR_B1_PMON_EVNT_SEL3	
06_2EH	See Table 2-17
MSR_B1_PMON_MASK	
06_2EH	See Table 2-17
MSR_B1_PMON_MATCH	
06_2EH	See Table 2-17
MSR_BBL_CR_CTL	
06_09H	See Table 2-59
MSR_BBL_CR_CTL3	
06_0FH, 06_17H	
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_0EH	See Table 2-58
06_09H	See Table 2-59
MSR_BIOS_DEBUG	
06_8CH, 06_8DH	See Table 2-45
MSR_BIOS_DONE	
06_7DH, 06_7EH	See Table 2-44
MSR_BIOS_MCU_ERRORCODE	
06_7DH, 06_7EH	See Table 2-44
06_8CH, 06_8DH	See Table 2-45
MSR_BPU_CCCRO	
0FH	See Table 2-55
MSR_BPU_CCCR1	
0FH	See Table 2-55
MSR_BPU_CCCR2	
0FH	See Table 2-55
MSR_BPU_CCCR3	
0FH	See Table 2-55
MSR_BPU_COUNTER0	
0FH	See Table 2-55
MSR_BPU_COUNTER1	
0FH	See Table 2-55
MSR_BPU_COUNTER2	
0FH	See Table 2-55
MSR_BPU_COUNTER3	
0FH	See Table 2-55
MSR_BPU_ESCR0	
0FH	See Table 2-55
MSR_BPU_ESCR1	
0FH	See Table 2-55
MSR_BR_DETECT_COUNTER_CONFIG_i	
06_66H	See Table 2-42
MSR_BR_DETECT_CTRL	
06_66H	See Table 2-42
MSR_BR_DETECT_STATUS	
06_66H	See Table 2-42
MSR_BSU_ESCR0	
0FH	See Table 2-55
MSR_BSU_ESCR1	
0FH	See Table 2-55
MSR_CO_PMON_BOX_CTRL	
06_2EH	See Table 2-17
06_2DH	See Table 2-24

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_3FH	See Table 2-33
MSR_CO_PMON_BOX_FILTER	
06_2DH	See Table 2-24
MSR_CO_PMON_BOX_FILTERO	
06_3FH	See Table 2-33
MSR_CO_PMON_BOX_FILTER1	
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_CO_PMON_BOX_OVF_CTRL	
06_2EH	See Table 2-17
MSR_CO_PMON_BOX_STATUS	
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_CO_PMON_CTRO	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_CO_PMON_CTR1	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_CO_PMON_CTR2	
06_2EH	
06_2DH	
06_3FH	See Table 2-33
MSR_CO_PMON_CTR3	
06_2EH	See Table 2-17
06_2DH	
06_3FH	See Table 2-33
MSR_CO_PMON_CTR4	
06_2EH	See Table 2-17
MSR_CO_PMON_CTR5	
06_2EH	See Table 2-17
MSR_CO_PMON_EVNT_SEL0	
06_2EH	
06_2DH	
06_3FH	See Table 2-33
MSR_CO_PMON_EVNT_SEL1	
06_2EH	
06_2DH	
06_3FH	See Table 2-33
MSR_CO_PMON_CTR1	

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_CO_PMON_EVNT_SEL2	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_CO_PMON_CTR2	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_CO_PMON_EVNT_SEL3	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_CO_PMON_EVNT_SEL4	
06_2EH	See Table 2-17
MSR_CO_PMON_EVNT_SEL5	
06_2EH	See Table 2-17
MSR_C1_PMON_BOX_CTRL	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C1_PMON_BOX_FILTER	
06_2DH	See Table 2-24
MSR_C1_PMON_BOX_FILTER0	
06_3FH	See Table 2-33
MSR_C1_PMON_BOX_FILTER1	
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C1_PMON_BOX_OVF_CTRL	
06_2EH	See Table 2-17
MSR_C1_PMON_BOX_STATUS	
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_C1_PMON_CTR0	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C1_PMON_CTR1	
06_2EH	See Table 2-17
06_2DH	See Table 2-24

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_3FH	See Table 2-33
MSR_C1_PMON_CTR2	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C1_PMON_CTR3	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C1_PMON_CTR4	
06_2EH	See Table 2-17
MSR_C1_PMON_CTR5	
06_2EH	See Table 2-17
MSR_C1_PMON_EVNT_SEL0	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C1_PMON_EVNT_SEL1	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C1_PMON_EVNT_SEL2	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C1_PMON_EVNT_SEL3	
06_2EH	
06_2DH	
06_3FH	See Table 2-33
MSR_C1_PMON_EVNT_SEL4	
06_2EH	See Table 2-17
MSR_C1_PMON_EVNT_SEL5	
06_2EH	See Table 2-17
MSR_C10_PMON_BOX_FILTER	
06_3EH	See Table 2-28
MSR_C10_PMON_BOX_FILTER0	
06_3FH	See Table 2-33
MSR_C10_PMON_BOX_FILTER1	
06_3EH	
06_3FH	See Table 2-33
MSR_C11_PMON_BOX_FILTER	
06_3EH	See Table 2-28

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_C11_PMON_BOX_FILTER0	
06_3FH	See Table 2-33
MSR_C11_PMON_BOX_FILTER1	
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C12_PMON_BOX_FILTER	
06_3EH	See Table 2-28
MSR_C12_PMON_BOX_FILTER0	
06_3FH	See Table 2-33
MSR_C12_PMON_BOX_FILTER1	
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C13_PMON_BOX_FILTER	
06_3EH	See Table 2-28
MSR_C13_PMON_BOX_FILTER0	
06_3FH	See Table 2-33
MSR_C13_PMON_BOX_FILTER1	
	See Table 2-28
MSR_C14_PMON_BOX_FILTER	
06_3EH	See Table 2-28
MSR_C14_PMON_BOX_FILTER0	
06_3FH	See Table 2-33
MSR_C14_PMON_BOX_FILTER1	333 13310 2 33
06_3EH	See Table 2-28
06_3FH	
MSR_C15_PMON_BOX_CTL	300 10010 2 33
06_3FH	See Table 2-33
MSR_C15_PMON_BOX_FILTER0	See Table 2 33
06_3FH	See Table 2-33
MSR_C15_PMON_BOX_FILTER1	See Table 2 33
06_3FH	See Table 2-33
MSR_C15_PMON_BOX_STATUS	See Table 2 33
06_3FH	Soo Tahlo 2-33
MSR_C15_PMON_CTR0	See Table 2 33
06_3FH	Soo Table 2-33
MSR_C15_PMON_CTR1	See Table 2 33
06_3FH	Soo Table 2-33
MSR_C15_PMON_CTR2	2cc 10016 7-22
06_3FH	Soo Table 2-22
MSR_C15_PMON_CTR3	266 Janie 7-33
06_3FH	Soo Table 2 22
···	266 Janie 7-32

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_C15_PMON_EVNTSEL0	
06_3FH	See Table 2-33
MSR_C15_PMON_EVNTSEL1	
06_3FH	See Table 2-33
MSR_C15_PMON_EVNTSEL2	
06_3FH	See Table 2-33
MSR_C15_PMON_EVNTSEL3	
06_3FH	See Table 2-33
MSR_C16_PMON_BOX_CTL	
06_3FH	See Table 2-33
MSR_C16_PMON_BOX_FILTER0	
06_3FH	See Table 2-33
MSR_C16_PMON_BOX_FILTER1	
06_3FH	See Table 2-33
MSR_C16_PMON_BOX_STATUS	
06_3FH	See Table 2-33
MSR_C16_PMON_CTR0	
06_3FH	See Table 2-33
MSR_C16_PMON_CTR3	
06_3FH	See Table 2-33
MSR_C16_PMON_CTR2	
06_3FH	See Table 2-33
MSR_C16_PMON_CTR3	
06_3FH	See Table 2-33
MSR_C16_PMON_EVNTSEL0	
06_3FH	See Table 2-33
MSR_C16_PMON_EVNTSEL1	
06_3FH	See Table 2-33
MSR_C16_PMON_EVNTSEL2	
06_3FH	See Table 2-33
MSR_C16_PMON_EVNTSEL3	
06_3FH	See Table 2-33
MSR_C17_PMON_BOX_CTL	
06_3FH	See Table 2-33
MSR_C17_PMON_BOX_FILTER0	
06_3FH	See Table 2-33
MSR_C17_PMON_BOX_FILTER1	
06_3FH	See Table 2-33
MSR_C17_PMON_BOX_STATUS	
06_3FH	See Table 2-33
MSR_C17_PMON_CTR0	
06_3FH	See Table 2-33

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_C17_PMON_CTR1	
06_3FH	See Table 2-33
MSR_C17_PMON_CTR2	
06_3FH	See Table 2-33
MSR_C17_PMON_CTR3	
06_3FH	See Table 2-33
MSR_C17_PMON_EVNTSEL0	
06_3FH	See Table 2-33
MSR_C17_PMON_EVNTSEL1	
06_3FH	See Table 2-33
MSR_C17_PMON_EVNTSEL2	
06_3FH	See Table 2-33
MSR_C17_PMON_EVNTSEL3	
06_3FH	See Table 2-33
MSR_C2_PMON_BOX_CTRL	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C2_PMON_BOX_FILTER	
06_2DH	See Table 2-24
MSR_C2_PMON_BOX_FILTER0	
06_3FH	See Table 2-33
MSR_C2_PMON_BOX_FILTER1	
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C2_PMON_BOX_OVF_CTRL	
06_2EH	See Table 2-17
MSR_C2_PMON_BOX_STATUS	
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_C2_PMON_CTR0	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C2_PMON_CTR1	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C2_PMON_CTR2	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_C2_PMON_CTR3	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C2_PMON_CTR4	
06_2EH	See Table 2-17
MSR_C2_PMON_CTR5	
06_2EH	See Table 2-17
MSR_C2_PMON_EVNT_SEL0	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C2_PMON_EVNT_SEL1	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C2_PMON_EVNT_SEL2	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C2_PMON_EVNT_SEL3	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C2_PMON_EVNT_SEL4	
06_2EH	See Table 2-17
MSR_C2_PMON_EVNT_SEL5	
06_2EH	See Table 2-17
MSR_C3_PMON_BOX_CTRL	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C3_PMON_BOX_FILTER	
06_2DH	See Table 2-24
MSR_C3_PMON_BOX_FILTER0	
06_3FH	See Table 2-33
MSR_C3_PMON_BOX_FILTER1	
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C3_PMON_BOX_OVF_CTRL	
06_2EH	See Table 2-17
MSR_C3_PMON_BOX_STATUS	

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_C3_PMON_CTR0	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C3_PMON_CTR1	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C3_PMON_CTR2	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C3_PMON_CTR3	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C3_PMON_CTR4	
06_2EH	See Table 2-17
MSR_C3_PMON_CTR5	
06_2EH	See Table 2-17
MSR_C3_PMON_EVNT_SEL0	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C3_PMON_EVNT_SEL1	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C3_PMON_EVNT_SEL2	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C3_PMON_EVNT_SEL3	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C3_PMON_EVNT_SEL4	
06_2EH	See Table 2-17
MSR_C3_PMON_EVNT_SEL5	
06_2EH	See Table 2-17

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_C4_PMON_BOX_CTRL	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C4_PMON_BOX_FILTER	
06_2DH	See Table 2-24
MSR_C4_PMON_BOX_FILTER0	
06_3FH	See Table 2-33
MSR_C4_PMON_BOX_FILTER1	
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C4_PMON_BOX_OVF_CTRL	
06_2EH	See Table 2-17
MSR_C4_PMON_BOX_STATUS	
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_C4_PMON_CTR0	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C4_PMON_CTR1	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C4_PMON_CTR2	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C4_PMON_CTR3	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C4_PMON_CTR4	
06_2EH	See Table 2-17
MSR_C4_PMON_CTR5	
06_2EH	See Table 2-17
MSR_C4_PMON_EVNT_SEL0	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C4_PMON_EVNT_SEL1	
06_2EH	See Table 2-17

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C4_PMON_EVNT_SEL2	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C4_PMON_EVNT_SEL3	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C4_PMON_EVNT_SEL4	
06_2EH	See Table 2-17
MSR_C4_PMON_EVNT_SEL5	
06_2EH	See Table 2-17
MSR_C5_PMON_BOX_CTRL	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C5_PMON_BOX_FILTER	
06_2DH	See Table 2-24
MSR_C5_PMON_BOX_FILTER0	
06_3FH	See Table 2-33
MSR_C5_PMON_BOX_FILTER1	
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C5_PMON_BOX_OVF_CTRL	
06_2EH	See Table 2-17
MSR_C5_PMON_BOX_STATUS	
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_C5_PMON_CTR0	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C5_PMON_CTR1	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C5_PMON_CTR2	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_C5_PMON_CTR3	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C5_PMON_CTR4	
06_2EH	See Table 2-17
MSR_C5_PMON_CTR5	
06_2EH	See Table 2-17
MSR_C5_PMON_EVNT_SEL0	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C5_PMON_EVNT_SEL1	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C5_PMON_EVNT_SEL2	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C5_PMON_EVNT_SEL3	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C5_PMON_EVNT_SEL4	
06_2EH	See Table 2-17
MSR_C5_PMON_EVNT_SEL5	
06_2EH	See Table 2-17
MSR_C6_PMON_BOX_CTRL	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C6_PMON_BOX_FILTER	
06_2DH	See Table 2-24
MSR_C6_PMON_BOX_FILTER0	
06_3FH	See Table 2-33
MSR_C6_PMON_BOX_FILTER1	
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C6_PMON_BOX_OVF_CTRL	
06_2EH	See Table 2-17
MSR_C6_PMON_BOX_STATUS	

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_C6_PMON_CTR0	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C6_PMON_CTR1	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C6_PMON_CTR2	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C6_PMON_CTR3	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C6_PMON_CTR4	
06_2EH	See Table 2-17
MSR_C6_PMON_CTR5	
06_2EH	See Table 2-17
MSR_C6_PMON_EVNT_SEL0	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C6_PMON_EVNT_SEL1	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	
MSR_C6_PMON_EVNT_SEL2	
06_2EH	See Table 2-17
MSR_C6_PMON_EVNT_SEL3	
06_2EH	See Table 2-17
06_2DH	
06_3FH	
MSR_C6_PMON_EVNT_SEL4	
	See Table 2-17
MSR_C6_PMON_EVNT_SEL5	
06_2EH	See Table 2-17

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_C7_PMON_BOX_CTRL	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C7_PMON_BOX_FILTER	
06_2DH	See Table 2-24
MSR_C7_PMON_BOX_FILTER0	
06_3FH	See Table 2-33
MSR_C7_PMON_BOX_FILTER1	
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C7_PMON_BOX_OVF_CTRL	
06_2EH	See Table 2-17
MSR_C7_PMON_BOX_STATUS	
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_C7_PMON_CTR0	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C7_PMON_CTR1	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C7_PMON_CTR2	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C7_PMON_CTR3	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C7_PMON_CTR4	
06_2EH	See Table 2-17
MSR_C7_PMON_CTR5	
06_2EH	See Table 2-17
MSR_C7_PMON_EVNT_SEL0	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C7_PMON_EVNT_SEL1	
06_2EH	See Table 2-17

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C7_PMON_EVNT_SEL2	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C7_PMON_EVNT_SEL3	
06_2EH	See Table 2-17
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_C7_PMON_EVNT_SEL4	
06_2EH	See Table 2-17
MSR_C7_PMON_EVNT_SEL5	
06_2EH	See Table 2-17
MSR_C8_PMON_BOX_CTRL	
06_2FH	See Table 2-19
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C8_PMON_BOX_FILTER	
06_3EH	See Table 2-28
MSR_C8_PMON_BOX_FILTER0	
06_3FH	See Table 2-33
MSR_C8_PMON_BOX_FILTER1	
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C8_PMON_BOX_OVF_CTRL	
06_2FH	See Table 2-19
MSR_C8_PMON_BOX_STATUS	
06_2FH	See Table 2-19
06_3FH	See Table 2-33
MSR_C8_PMON_CTR0	
06_2FH	See Table 2-19
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C8_PMON_CTR1	
06_2FH	See Table 2-19
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C8_PMON_CTR2	
06_2FH	See Table 2-19
06_3EH	See Table 2-28
06_3FH	See Table 2-33

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_C8_PMON_CTR3	
06_2FH	See Table 2-19
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C8_PMON_CTR4	
06_2FH	See Table 2-19
MSR_C8_PMON_CTR5	
06_2FH	See Table 2-19
MSR_C8_PMON_EVNT_SEL0	
06_2FH	See Table 2-19
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C8_PMON_EVNT_SEL1	
06_2FH	See Table 2-19
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C8_PMON_EVNT_SEL2	
06_2FH	See Table 2-19
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C8_PMON_EVNT_SEL3	
06_2FH	See Table 2-19
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C8_PMON_EVNT_SEL4	
06_2FH	See Table 2-19
MSR_C8_PMON_EVNT_SEL5	
06_2FH	See Table 2-19
MSR_C9_PMON_BOX_CTRL	
06_2FH	See Table 2-19
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C9_PMON_BOX_FILTER	
06_3EH	See Table 2-28
MSR_C9_PMON_BOX_FILTER0	
06_3FH	See Table 2-33
MSR_C9_PMON_BOX_FILTER1	
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C9_PMON_BOX_OVF_CTRL	
06_2FH	See Table 2-19
MSR_C9_PMON_BOX_STATUS	

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_2FH	See Table 2-19
06_3FH	See Table 2-33
MSR_C9_PMON_CTR0	
06_2FH	See Table 2-19
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C9_PMON_CTR1	
06_2FH	See Table 2-19
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C9_PMON_CTR2	
06_2FH	See Table 2-19
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C9_PMON_CTR3	
06_2FH	See Table 2-19
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C9_PMON_CTR4	
06_2FH	See Table 2-19
MSR_C9_PMON_CTR5	
06_2FH	See Table 2-19
MSR_C9_PMON_EVNT_SEL0	
06_2FH	See Table 2-19
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C9_PMON_EVNT_SEL1	
06_2FH	See Table 2-19
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C9_PMON_EVNT_SEL2	
06_2FH	See Table 2-19
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C9_PMON_EVNT_SEL3	
06_2FH	See Table 2-19
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_C9_PMON_EVNT_SEL4	
06_2FH	See Table 2-19
MSR_C9_PMON_EVNT_SEL5	
06_2FH	See Table 2-19

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_CC6_DEMOTION_POLICY_CONFIG	
06_37H	See Table 2-9
MSR_CONFIG_TDP_CONTROL	
06_3AH	See Table 2-25
06_3CH, 06_45H, 06_46H	See Table 2-29
06_57H, 06_85H	See Table 2-53
MSR_CONFIG_TDP_LEVEL1	
06_3AH	See Table 2-25
06_3CH, 06_45H, 06_46H	See Table 2-29
06_57H, 06_85H	See Table 2-53
MSR_CONFIG_TDP_LEVEL2	
06_3AH	See Table 2-25
06_3CH, 06_45H, 06_46H	See Table 2-29
06_57H, 06_85H	See Table 2-53
MSR_CONFIG_TDP_NOMINAL	
06_3AH	See Table 2-25
06_3CH, 06_45H, 06_46H	See Table 2-29
06_57H, 06_85H	See Table 2-53
MSR_CORE_C1_RESIDENCY	
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
06_66H	See Table 2-42
MSR_CORE_C3_RESIDENCY	
06_5CH, 06_7AH	See Table 2-12
06_1AH, 06_1EH, 06_1FH, 06_2EH, 06_25H, 06_2CH, 06_2FH	See Table 2-15
06_2AH, 06_2DH, 06_3AH, 06_3CH, 06_3EH, 06_3FH, 06_45H, 06_46H	See Table 2-20
MSR_CORE_C6_RESIDENCY	
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
06_1AH, 06_1EH, 06_1FH, 06_2EH, 06_25H, 06_2CH, 06_2FH	See Table 2-15
06_2AH, 06_2DH, 06_3AH, 06_3CH, 06_3EH, 06_3FH, 06_45H, 06_46H	See Table 2-20
06_57H, 06_85H	See Table 2-53
MSR_CORE_C7_RESIDENCY	
06_2AH, 06_2DH, 06_3AH, 06_3CH, 06_3EH, 06_3FH, 06_45H, 06_46H	See Table 2-20
MSR_CORE_GFXE_OVERLAP_CO	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_9 06_6AH, 06_6CH	
MSR_CORE_HDC_RESIDENCY	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_9 06_6AH, 06_6CH	
MSR_CORE_PERF_LIMIT_REASONS	
06_5CH, 06_7AH	See Table 2-12
06_3CH, 06_45H, 06_46H	See Table 2-30
06_3F	See Table 2-32

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_56H, 06_4FH	See Table 2-36
06_57H, 06_85H	See Table 2-53
MSR_CORE_THREAD_COUNT	
06_3FH	See Table 2-32
MSR_CRASHLOG_CONTROL	
06_7DH, 06_7EH	See Table 2-44
MSR_CRU_ESCRO	
0FH	See Table 2-55
MSR_CRU_ESCR1	
0FH	See Table 2-55
MSR_CRU_ESCR2	
0FH	See Table 2-55
MSR_CRU_ESCR3	
0FH	See Table 2-55
MSR_CRU_ESCR4	
0FH	See Table 2-55
MSR_CRU_ESCR5	
0FH	See Table 2-55
MSR_DAC_ESCRO	
0FH	See Table 2-55
MSR_DAC_ESCR1	
0FH	See Table 2-55
MSR_DRAM_ENERGY_ STATUS	
06_5CH, 06_7AH	See Table 2-12
06_2DH	See Table 2-23
06_3EH, 06_3FH	See Table 2-26
06_3CH, 06_45H, 06_46H	See Table 2-29
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-36
06_6AH, 06_6CH	See Table 2-51
06_57H, 06_85H	See Table 2-53
MSR_DRAM_PERF_STATUS	
06_5CH, 06_7AH	See Table 2-12
06_2DH	See Table 2-23
06_3EH, 06_3FH	See Table 2-26
06_3CH, 06_45H, 06_46H	See Table 2-29
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-36
06_6AH, 06_6CH	See Table 2-51
06_57H, 06_85H	See Table 2-53
MSR_DRAM_POWER_INFO	
06_5CH, 06_7AH	See Table 2-12

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_2DH	See Table 2-23
06_3EH, 06_3FH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-36
06_6AH, 06_6CH	See Table 2-51
06_57H, 06_85H	See Table 2-53
MSR_DRAM_POWER_LIMIT	
06_5CH, 06_7AH	See Table 2-12
06_2DH	See Table 2-23
06_3EH, 06_3FH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-36
06_6AH, 06_6CH	See Table 2-51
06_57H, 06_85H	See Table 2-53
MSR_EBC_FREQUENCY_ID	
0FH	See Table 2-55
MSR_EBC_HARD_POWERON	
0FH	See Table 2-55
MSR_EBC_SOFT_POWERON	
0FH	See Table 2-55
MSR_EBL_CR_POWERON	
06_0FH, 06_17H	See Table 2-3
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
06_0EH	See Table 2-58
06_09H	See Table 2-59
MSR_EFSB_DRDY0	
0F_03H, 0F_04H	See Table 2-56
MSR_EFSB_DRDY1	
0F_03H, 0F_04H	See Table 2-56
MSR_EMON_L3_CTR_CTL0	
06_0FH, 06_17H	See Table 2-3
0F_06H	See Table 2-57
MSR_EMON_L3_CTR_CTL1	
06_0FH, 06_17H	See Table 2-3
0F_06H	See Table 2-57
MSR_EMON_L3_CTR_CTL2	
06_0FH, 06_17H	See Table 2-3
0F_06H	See Table 2-57
MSR_EMON_L3_CTR_CTL3	
06_0FH, 06_17H	See Table 2-3
0F_06H	See Table 2-57

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_EMON_L3_CTR_CTL4	
06_0FH, 06_17H	See Table 2-3
0F_06H	See Table 2-57
MSR_EMON_L3_CTR_CTL5	
06_0FH, 06_17H	See Table 2-3
0F_06H	See Table 2-57
MSR_EMON_L3_CTR_CTL6	
06_0FH, 06_17H	See Table 2-3
0F_06H	See Table 2-57
MSR_EMON_L3_CTR_CTL7	
06_0FH, 06_17H	See Table 2-3
0F_06H	See Table 2-57
MSR_EMON_L3_GL_CTL	
06_0FH, 06_17H	See Table 2-3
MSR_ERROR_CONTROL	
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
MSR_FAST_UNCORE_MSRS_CAPABILITY	
06_7DH, 06_7EH	See Table 2-44
MSR_FAST_UNCORE_MSRS_CTL	
06_7DH, 06_7EH	See Table 2-44
MSR_FAST_UNCORE_MSRS_STATUS	
06_7DH, 06_7EH	See Table 2-44
MSR_FEATURE_CONFIG	
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
06_25H, 06_2CH	See Table 2-18
06_2FH	See Table 2-19
06_2AH, 06_2DH	See Table 2-20
06_57H, 06_85H	See Table 2-53
MSR_FIRM_ESCR0	
0FH	See Table 2-55
MSR_FIRM_ESCR1	
0FH	See Table 2-55
MSR_FLAME_CCCR0	
0FH	See Table 2-55
MSR_FLAME_CCCR1	
0FH	See Table 2-55
MSR_FLAME_CCCR2	
0FH	See Table 2-55
MSR_FLAME_CCCR3	
0FH	See Table 2-55

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_FLAME_COUNTERO	
0FH	See Table 2-55
MSR_FLAME_COUNTER1	
0FH	See Table 2-55
MSR_FLAME_COUNTER2	
0FH	See Table 2-55
MSR_FLAME_COUNTER3	
0FH	See Table 2-55
MSR_FLAME_ESCR0	
0FH	See Table 2-55
MSR_FLAME_ESCR1	
0FH	See Table 2-55
MSR_FSB_ESCR0	
0FH	See Table 2-55
MSR_FSB_ESCR1	
0FH	See Table 2-55
MSR_FSB_FREQ	
06_0FH, 06_17H	
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	
06_4CH	
06_0EH	See Table 2-58
MSR_GQ_SN00P_MESF	Soo Table 2 16
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-10
06_3CH, 06_45H, 06_46H	Soo Table 2 20
MSR_IFSB_BUSQ0	See Table 2-30
0F 03H, 0F 04H	Soo Tahlo 2-56
MSR_IFSB_BUSQ1	Jee Tuble 2 Ju
0F_03H, 0F_04H	See Table 2-56
MSR_IFSB_CNTR7	300 105/0 2 30
0F_03H, 0F_04H	See Table 2-56
MSR IFSB CTL6	
OF_03H, 0F_04H	See Table 2-56
MSR_IFSB_SNPQ0	
0F_03H, 0F_04H	See Table 2-56
MSR_IFSB_SNPQ1	
0F_03H, 0F_04H	See Table 2-56
MSR_IQ_CCCR0	
0FH	See Table 2-55
MSR_IQ_CCCR1	
0eh	See Table 2-55

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_IQ_CCCR2	
0FH	See Table 2-55
MSR_IQ_CCCR3	
0FH	See Table 2-55
MSR_IQ_CCCR4	
0FH	See Table 2-55
MSR_IQ_CCCR5	
0FH	See Table 2-55
MSR_IQ_COUNTER0	
0FH	See Table 2-55
MSR_IQ_COUNTER1	
0FH	See Table 2-55
MSR_IQ_COUNTER2	
0FH	See Table 2-55
MSR_IQ_COUNTER3	
0FH	See Table 2-55
MSR_IQ_COUNTER4	
0FH	See Table 2-55
MSR_IQ_COUNTER5	
0FH	See Table 2-55
MSR_IQ_ESCR0	
0FH	See Table 2-55
MSR_IQ_ESCR1	
0FH	See Table 2-55
MSR_IS_ESCR0	
0FH	See Table 2-55
MSR_IS_ESCR1	
0FH	See Table 2-55
MSR_ITLB_ESCR0	
0FH	See Table 2-55
MSR_ITLB_ESCR1	
0FH	See Table 2-55
MSR_IX_ESCR0	
0FH	See Table 2-55
MSR_IX_ESCR1	
0FH	See Table 2-55
MSR_LASTBRANCH_0	
0FH	
06_0EH	
06_09H	See Table 2-59
MSR_LASTBRANCH_0_FROM_IP	
06_0FH, 06_17H	See Table 2-3

MSR Nam	e and CPUID DisplayFamily_DisplayModel	Location
	06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
	06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
	06_5CH	See Table 2-12
	06_7AH	See Table 2-13
	06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
	06_2AH, 06_2DH	See Table 2-20
	0FH	See Table 2-55
MSR_LAS	TBRANCH_O_TO_IP	
	06_0FH, 06_17H	See Table 2-3
	06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
	06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
	06_5CH	See Table 2-12
	06_7AH	See Table 2-13
	06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
	06_2AH, 06_2DH	See Table 2-20
	0FH	See Table 2-55
MSR_LAS	TBRANCH_1_FROM_IP	
	06_0FH, 06_17H	See Table 2-3
	06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
	06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
	06_5CH, 06_7AH	See Table 2-12
	06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
	06_2AH, 06_2DH	See Table 2-20
	0FH	See Table 2-55
MSR_LAS	TBRANCH_1_TO_IP	
	06_0FH, 06_17H	See Table 2-3
	06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
	06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
	06_5CH, 06_7AH	See Table 2-12
	06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
	06_2AH, 06_2DH	See Table 2-20
	0FH	See Table 2-55
MSR_LAS	TBRANCH_10_FROM_IP	
	06_5CH, 06_7AH	See Table 2-12
	06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
	06_2AH, 06_2DH	See Table 2-20
	0FH	See Table 2-55
MSR_LAS	TBRANCH_10_T0_IP	
	06_5CH, 06_7AH	See Table 2-12
	06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
	06_2AH, 06_2DH	See Table 2-20
	0FH	See Table 2-55

MSR Name ar	nd CPUID DisplayFamily_DisplayModel	Location
MSR_LASTBR	ANCH_11_FROM_IP	
06	5_5CH, 06_7AH	See Table 2-12
06	5_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06	5_2AH, 06_2DH	See Table 2-20
Of	H	See Table 2-55
MSR_LASTBR	ANCH_11_TO_IP	
06	5_5CH, 06_7AH	See Table 2-12
06	5_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06	5_2AH, 06_2DH	See Table 2-20
Of	- H	See Table 2-55
MSR_LASTBR	ANCH_12_FROM_IP	
06	5_5CH, 06_7AH	See Table 2-12
06	5_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06	5_2AH, 06_2DH	See Table 2-20
Of	H	See Table 2-55
MSR_LASTBR	ANCH_12_TO_IP	
06	5_5CH, 06_7AH	See Table 2-12
06	5_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06	5_2AH, 06_2DH	See Table 2-20
Of	H	See Table 2-55
MSR_LASTBR	ANCH_13_FROM_IP	
06	5_5CH, 06_7AH	See Table 2-12
06	5_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06	5_2AH, 06_2DH	See Table 2-20
Of	H	See Table 2-55
MSR_LASTBR	anch_13_to_ip	
06	5_5CH, 06_7AH	See Table 2-12
06	5_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06	5_2AH, 06_2DH	See Table 2-20
Of	₩	See Table 2-55
MSR_LASTBR	ANCH_14_FROM_IP	
06	5_5CH, 06_7AH	See Table 2-12
06	5_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06	5_2AH, 06_2DH	See Table 2-20
Of	H	See Table 2-55
MSR_LASTBR	ANCH_14_TO_IP	
06	5_5CH, 06_7AH	See Table 2-12
06	5_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06	5_2AH, 06_2DH	See Table 2-20
Of	H	See Table 2-55
MSR_LASTBR	ANCH_15_FROM_IP	
06	5_5CH, 06_7AH	See Table 2-12

MSR Name	e and CPUID DisplayFamily_DisplayModel	Location
	06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
	06_2AH, 06_2DH	See Table 2-20
	0FH	See Table 2-55
MSR_LAST	Branch_15_to_ip	
	06_5CH, 06_7AH	See Table 2-12
	06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
	06_2AH, 06_2DH	See Table 2-20
	0FH	See Table 2-55
MSR_LAST	Branch_16_from_ip	
	06_5CH, 06_7AH	See Table 2-12
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LAST	Branch_16_to_ip	
	06_5CH, 06_7AH	See Table 2-12
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LAST	Branch_17_from_ip	
	06_5CH, 06_7AH	See Table 2-12
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LAST	Branch_17_to_ip	
	06_5CH, 06_7AH	See Table 2-12
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LAST	Branch_18_from_ip	
	06_5CH, 06_7AH	See Table 2-12
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LAST	Branch_18_to_ip	
	06_5CH, 06_7AH	See Table 2-12
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LAST	BRANCH_19_FROM_IP	
	06_5CH, 06_7AH	See Table 2-12
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LAST	Branch_19_to_ip	
	06_5CH, 06_7AH	See Table 2-12
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LAST	BRANCH_2	
	0FH	See Table 2-55
	06_0EH	See Table 2-58
	06_09H	See Table 2-59

MSR Name	e and CPUID DisplayFamily_DisplayModel	Location
MSR_LAST	Branch_2_from_ip	
	06_0FH, 06_17H	See Table 2-3
	06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
	06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
	06_5CH, 06_7AH	See Table 2-12
	06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
	06_2AH, 06_2DH	See Table 2-20
	0FH	See Table 2-55
MSR_LAST	Branch_2_to_ip	
	06_0FH, 06_17H	See Table 2-3
	06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
	06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
	06_5CH, 06_7AH	See Table 2-12
	06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
	06_2AH, 06_2DH	See Table 2-20
	0FH	See Table 2-55
MSR_LAST	Branch_20_from_ip	
	06_5CH, 06_7AH	See Table 2-12
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LAST	Branch_20_to_ip	
	06_5CH, 06_7AH	See Table 2-12
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LAST	Branch_21_from_ip	
	06_5CH, 06_7AH	See Table 2-12
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LAST	Branch_21_to_ip	
	06_5CH, 06_7AH	See Table 2-12
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LAST	Branch_22_from_ip	
	06_5CH, 06_7AH	See Table 2-12
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LAST	Branch_22_to_ip	
	06_5CH, 06_7AH	See Table 2-12
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LAST	Branch_23_from_ip	
	06_5CH, 06_7AH	See Table 2-12
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_LASTBRANCH_23_TO_IP	
06_5CH, 06_7AH	See Table 2-12
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LASTBRANCH_24_FROM_IP	
06_5CH, 06_7AH	See Table 2-12
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LASTBRANCH_24_TO_IP	
06_5CH, 06_7AH	See Table 2-12
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	
MSR_LASTBRANCH_25_FROM_IP	
06_5CH, 06_7AH	See Table 2-12
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LASTBRANCH_25_TO_IP	
06_5CH, 06_7AH	See Table 2-12
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LASTBRANCH_26_FROM_IP	
06_5CH, 06_7AH	See Table 2-12
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	
MSR_LASTBRANCH_26_TO_IP	
06_5CH, 06_7AH	See Table 2-12
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LASTBRANCH_27_FROM_IP	
06_5CH, 06_7AH	See Table 2-12
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LASTBRANCH_27_TO_IP	
06_5CH, 06_7AH	See Table 2-12
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LASTBRANCH_28_FROM_IP	
06_5CH, 06_7AH	See Table 2-12
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_LASTBRANCH_28_TO_IP	
06_5CH, 06_7AH	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	
MSR_LASTBRANCH_29_FROM_IP	

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_5CH, 06_7AH	See Table 2-1 <i>2</i>
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH 06_6AH, 06_6CH	
MSR_LASTBRANCH_29_TO_IP	
06_5CH, 06_7AH	See Table 2-12
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH 06_6AH, 06_6CH	
MSR_LASTBRANCH_3	
0FH	See Table 2-55
06_0EH	See Table 2-58
06_09H	See Table 2-59
MSR_LASTBRANCH_3_FROM_IP	
06_0FH, 06_17H	See Table 2-3
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
06_5CH, 06_7AH	See Table 2-12
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	See Table 2-20
0FH	See Table 2-55
MSR_LASTBRANCH_3_TO_IP	
06_0FH, 06_17H	See Table 2-3
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
06_5CH, 06_7AH	See Table 2-12
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	See Table 2-20
0FH	See Table 2-55
MSR_LASTBRANCH_30_FROM_IP	
06_5CH, 06_7AH	See Table 2-12
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH 06_6AH, 06_6CH	
MSR_LASTBRANCH_30_T0_IP	
06_5CH, 06_7AH	See Table 2-12
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH 06_6AH, 06_6CH	
MSR_LASTBRANCH_31_FROM_IP	
06_5CH, 06_7AH	See Table 2-12
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH 06_6AH, 06_6CH	
MSR_LASTBRANCH_31_TO_IP	
06_5CH, 06_7AH	See Table 2-12
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH	
06_6AH, 06_6CH	

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_LASTBRANCH_4	
06_0EH	See Table 2-58
06_09H	See Table 2-59
MSR_LASTBRANCH_4_FROM_IP	
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
06_5CH, 06_7AH	See Table 2-12
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	See Table 2-20
0FH	See Table 2-55
MSR_LASTBRANCH_4_TO_IP	
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
06_5CH, 06_7AH	See Table 2-12
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	See Table 2-20
0FH	See Table 2-55
MSR_LASTBRANCH_5	
06_0EH	See Table 2-58
06_09H	See Table 2-59
MSR_LASTBRANCH_5_FROM_IP	
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
06_5CH, 06_7AH	See Table 2-12
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	See Table 2-20
0FH	See Table 2-55
MSR_LASTBRANCH_5_TO_IP	
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
06_5CH, 06_7AH	See Table 2-12
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	See Table 2-20
0FH	See Table 2-55
MSR_LASTBRANCH_6	
06_0EH	See Table 2-58
06_09H	See Table 2-59
MSR_LASTBRANCH_6_FROM_IP	
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
06_5CH, 06_7AH	See Table 2-12
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_2AH, 06_2DH	See Table 2-20
0FH	See Table 2-55
MSR_LASTBRANCH_6_TO_IP	
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
06_5CH, 06_7AH	See Table 2-12
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	See Table 2-20
0FH	See Table 2-55
MSR_LASTBRANCH_7	
06_0EH	See Table 2-58
06_09H	See Table 2-59
MSR_LASTBRANCH_7_FROM_IP	
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
06_5CH, 06_7AH	See Table 2-12
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	See Table 2-20
0FH	See Table 2-55
MSR_LASTBRANCH_7_TO_IP	
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
06_5CH, 06_7AH	See Table 2-12
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	See Table 2-20
0FH	See Table 2-55
MSR_LASTBRANCH_8_FROM_IP	
06_5CH, 06_7AH	See Table 2-12
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	See Table 2-20
0FH	See Table 2-55
MSR_LASTBRANCH_8_TO_IP	
06_5CH, 06_7AH	See Table 2-12
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	See Table 2-20
0FH	See Table 2-55
MSR_LASTBRANCH_9_FROM_IP	
06_5CH, 06_7AH	See Table 2-12
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	See Table 2-20
0FH	See Table 2-55
MSR_LASTBRANCH_9_TO_IP	

MSR Name a	and CPUID DisplayFamily_DisplayModel	Location
0	06_5CH, 06_7AH	See Table 2-12
0	06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
0	06_2AH, 06_2DH	See Table 2-20
0)FH	See Table 2-55
MSR_LASTBF	RANCH_TOS	
0	06_0FH, 06_17H	See Table 2-3
0	06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
0	06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
0	06_5CH, 06_7AH	See Table 2-12
0	06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
0	06_2AH, 06_2DH	See Table 2-20
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
0	06_57H, 06_85H	See Table 2-53
0	06_0EH	See Table 2-58
0	06_09H	See Table 2-59
MSR_LASTBE	RANCH_INFO_0	
0	06_7AH	See Table 2-13
MSR_LBR_IN	F0_1	
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
0	06_7AH	See Table 2-13
MSR_LBR_IN	F0_10	
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
0	06_7AH	See Table 2-13
MSR_LBR_IN	F0_11	
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
0	06_7AH	See Table 2-13
MSR_LBR_IN	F0_12	
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
0	06_7AH	See Table 2-13
MSR_LBR_IN	F0_13	
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
0	06_7AH	See Table 2-13
MSR_LBR_IN	FO_14	
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
0	06_7AH	See Table 2-13
MSR LBR IN	FO 15	

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
06_7AH	See Table 2-13
MSR_LBR_INFO_16	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	
06_7AH	See Table 2-13
MSR_LBR_INFO_17	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	
06_7AH	See Table 2-13
MSR_LBR_INFO_18	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
06_7AH	See Table 2-13
MSR_LBR_INFO_19	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	
06_7AH	See Table 2-13
MSR_LBR_INFO_2	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	
06_7AH	See Table 2-13
MSR_LBR_INFO_20	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	
06_7AH	See Table 2-13
MSR_LBR_INFO_21	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	
06_7AH	See Table 2-13
MSR_LBR_INFO_22	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	
06_7AH	See Table 2-13
MSR_LBR_INFO_23	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
06_7AH	See Table 2-13
MSR_LBR_INFO_24	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	
 06_7АН	
MSR LRR INFO 25	

MSR Name	e and CPUID DisplayFamily_DisplayModel	Location
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
	06_7AH	See Table 2-13
MSR_LBR_	INFO_26	
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
	06_7AH	See Table 2-13
MSR_LBR_	INFO_27	
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
	06_7AH	See Table 2-13
MSR_LBR_	INFO_28	
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
	06_7AH	See Table 2-13
MSR_LBR_	INFO_29	
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
	06_7AH	See Table 2-13
MSR_LBR_	INFO_3	
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
	06_7AH	See Table 2-13
MSR_LBR_	INFO_30	
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
	06_7AH	See Table 2-13
MSR_LBR_	INFO_31	
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
	06_7AH	See Table 2-13
MSR_LBR_	INFO_4	
	06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
	06 7AH	See Table 2-13
MSR_LBR_	INFO 5	
		See Table 2-39
	 06_7AH	See Table 2-13
MSR_LBR_		•
	0_0 06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH,	See Table 2-39
	06_7AH	
MSR I BR	_	SEE IGNIE 7-12
I I DE LEDE	IIVI () /	

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
06_7AH	See Table 2-13
MSR_LBR_INFO_8	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
06_7AH	See Table 2-13
MSR_LBR_INFO_9	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
06_7AH	See Table 2-13
MSR_LBR_SELECT	
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
06_5CH, 06_7AH	
06_1AH, 06_1EH, 06_1FH, 06_2EH	
06_2AH, 06_2DH	
06_3CH, 06_45H, 06_46H	
_	
MSR_LER_FROM_LIP	
 06_0FH, 06_17H	See Table 2-3
 06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	
06_1AH, 06_1EH, 06_1FH, 06_2EH	
06_2AH, 06_2DH	
06_57H, 06_85H	
0FH	See Table 2-55
06_0EH	See Table 2-58
06_09H	See Table 2-59
MSR_LER_TO_LIP	
06_0FH, 06_17H	See Table 2-3
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
06_1AH, 06_1EH, 06_1FH, 06_2EH	
06_2AH, 06_2DH	See Table 2-20
06_57H, 06_85H	See Table 2-53
0FH	See Table 2-55
06_0EH	See Table 2-58
06_09H	See Table 2-59
MSR_MO_PMON_ADDR_MASK	
06_2EH	See Table 2-17
MSR_MO_PMON_ADDR_MATCH	
06_2EH	See Table 2-17
MSR_M0_PMON_BOX_CTRL	

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_2EH	See Table 2-17
MSR_M0_PMON_BOX_OVF_CTRL	
06_2EH	See Table 2-17
MSR_MO_PMON_BOX_STATUS	
06_2EH	See Table 2-17
MSR_M0_PMON_CTR0	
06_2EH	See Table 2-17
MSR_M0_PM0N_CTR1	
06_2EH	See Table 2-17
MSR_MO_PMON_CTR2	
06_2EH	See Table 2-17
MSR_MO_PMON_CTR3	
06_2EH	See Table 2-17
MSR_MO_PMON_CTR4	
06_2EH	See Table 2-17
MSR_MO_PMON_CTR5	
06_2EH	See Table 2-17
MSR_MO_PMON_DSP	
06_2EH	See Table 2-17
MSR_MO_PMON_EVNT_SELO	
06_2EH	See Table 2-17
MSR_MO_PMON_EVNT_SEL1	
06_2EH	See Table 2-17
MSR_MO_PMON_EVNT_SEL2	
06_2EH	See Table 2-17
MSR_MO_PMON_EVNT_SEL3	
06_2EH	See Table 2-17
MSR_MO_PMON_EVNT_SEL4	
06_2EH	See Table 2-17
MSR_MO_PMON_EVNT_SEL5	
06_2EH	See Table 2-17
MSR_MO_PMON_ISS	
06_2EH	See Table 2-17
MSR_MO_PMON_MAP	
06_2EH	See Table 2-17
MSR_MO_PMON_MM_CONFIG	
06_2EH	See Table 2-17
MSR_MO_PMON_MSC_THR	
06_2EH	See Table 2-17
MSR_MO_PMON_PGT	
06_2EH	See Table 2-17
MSR_MO_PMON_PLD	

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_2EH	See Table 2-17
MSR_MO_PMON_TIMESTAMP	
06_2EH	See Table 2-17
MSR_MO_PMON_ZDP	
06_2EH	See Table 2-17
MSR_M1_PMON_ADDR_MASK	
06_2EH	See Table 2-17
MSR_M1_PMON_ADDR_MATCH	
06_2EH	See Table 2-17
MSR_M1_PMON_BOX_CTRL	
06_2EH	See Table 2-17
MSR_M1_PMON_BOX_OVF_CTRL	
06_2EH	See Table 2-17
MSR_M1_PMON_BOX_STATUS	
06_2EH	See Table 2-17
MSR_M1_PMON_CTR0	
06_2EH	See Table 2-17
MSR_M1_PMON_CTR1	
06_2EH	See Table 2-17
MSR_M1_PMON_CTR2	
06_2EH	See Table 2-17
MSR_M1_PMON_CTR3	
06_2EH	See Table 2-17
MSR_M1_PMON_CTR4	
06_2EH	See Table 2-17
MSR_M1_PMON_CTR5	
06_2EH	See Table 2-17
MSR_M1_PMON_DSP	
06_2EH	See Table 2-17
MSR_M1_PMON_EVNT_SEL0	
06_2EH	See Table 2-17
MSR_M1_PMON_EVNT_SEL1	
06_2EH	See Table 2-17
MSR_M1_PMON_EVNT_SEL2	
06_2EH	See Table 2-17
MSR_M1_PMON_EVNT_SEL3	
06_2EH	See Table 2-17
MSR_M1_PMON_EVNT_SEL4	
06_2EH	See Table 2-17
MSR_M1_PMON_EVNT_SEL5	
06_2EH	See Table 2-17
MSR_M1_PMON_ISS	

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_2EH	See Table 2-17
MSR_M1_PMON_MAP	
06_2EH	See Table 2-17
MSR_M1_PMON_MM_CONFIG	
06_2EH	See Table 2-17
MSR_M1_PMON_MSC_THR	
06_2EH	See Table 2-17
MSR_M1_PMON_PGT	
06_2EH	See Table 2-17
MSR_M1_PMON_PLD	
06_2EH	See Table 2-17
MSR_M1_PMON_TIMESTAMP	
06_2EH	See Table 2-17
MSR_M1_PMON_ZDP	
06_2EH	See Table 2-17
IA32_MCO_MISC / MSR_MCO_MISC	
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
MSR_MCO_RESIDENCY	
06_57H, 06_85H	See Table 2-53
IA32_MC1_MISC / MSR_MC1_MISC	
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
IA32_MC10_ADDR / MSR_MC10_ADDR	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38
IA32_MC10_CTL / MSR_MC10_CTL	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38
IA32_MC10_MISC / MSR_MC10_MISC	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38

MSR Nam	e and CPUID DisplayFamily_DisplayModel	Location
IA32_MC1	IO_STATUS / MSR_MC10_STATUS	
	06_2EH	See Table 2-17
	06_2DH	See Table 2-23
	06_3EH	See Table 2-26
	06_3F	See Table 2-32
	06_56H, 06_4FH	See Table 2-37
	06_4FH	See Table 2-38
IA32_MC1	1_ADDR / MSR_MC11_ADDR	
	06_2EH	See Table 2-17
	06_2DH	See Table 2-23
	06_3EH	See Table 2-26
	06_3F	See Table 2-32
	06_4FH	See Table 2-38
IA32_MC1	11_CTL / MSR_MC11_CTL	
	06_2EH	See Table 2-17
	06_2DH	See Table 2-23
	06_3EH	See Table 2-26
	06_3F	See Table 2-32
	06_4FH	See Table 2-38
IA32_MC1	1_MISC / MSR_MC11_MISC	
	06_2EH	See Table 2-17
	06_2DH	See Table 2-23
	06_3EH	See Table 2-26
	06_3F	See Table 2-32
	06_4FH	See Table 2-38
IA32_MC1	1_STATUS / MSR_MC11_STATUS	
	06_2EH	See Table 2-17
	06_2DH	See Table 2-23
	06_3EH	See Table 2-26
	06_3F	See Table 2-32
	06_4FH	See Table 2-38
IA32_MC1	2_ADDR / MSR_MC12_ADDR	
	06_2EH	See Table 2-17
	06_2DH	See Table 2-23
	06_3EH	See Table 2-26
	06_3F	See Table 2-32
	06_4FH	See Table 2-38
IA32_MC1	2_CTL / MSR_MC12_CTL	
	06_2EH	See Table 2-17
	06_2DH	See Table 2-23
	06_3EH	See Table 2-26
	06_3F	See Table 2-32

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_4FH	See Table 2-38
IA32_MC12_MISC / MSR_MC12_MISC	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC12_STATUS / MSR_MC12_STATUS	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC13_ADDR / MSR_MC13_ADDR	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC13_CTL / MSR_MC13_CTL	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC13_MISC / MSR_MC13_MISC	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC13_STATUS / MSR_MC13_STATUS	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC14_ADDR / MSR_MC14_ADDR	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_4FH	See Table 2-38
IA32_MC14_CTL / MSR_MC14_CTL	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC14_MISC / MSR_MC14_MISC	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC14_STATUS / MSR_MC14_STATUS	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC15_ADDR / MSR_MC15_ADDR	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC15_CTL / MSR_MC15_CTL	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC15_MISC / MSR_MC15_MISC	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC15_STATUS / MSR_MC15_STATUS	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_4FH	See Table 2-38
IA32_MC16_ADDR / MSR_MC16_ADDR	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC16_CTL / MSR_MC16_CTL	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC16_MISC / MSR_MC16_MISC	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC16_STATUS / MSR_MC16_STATUS	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC17_ADDR / MSR_MC17_ADDR	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38
IA32_MC17_CTL / MSR_MC17_CTL	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38
IA32_MC17_MISC / MSR_MC17_MISC	
06_2EH	See Table 2-17
06_2DH	See Table 2-23

MODEL-SPECIFIC REGISTERS (MSRS)

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38
IA32_MC17_STATUS / MSR_MC17_STATUS	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38
IA32_MC18_ADDR / MSR_MC18_ADDR	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38
IA32_MC18_CTL / MSR_MC18_CTL	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38
IA32_MC18_MISC / MSR_MC18_MISC	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38
IA32_MC18_STATUS / MSR_MC18_STATUS	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	
IA32_MC19_ADDR / MSR_MC19_ADDR	
06_2EH	See Table 2-17
06_2DH	See Table 2-23

MSR Name	e and CPUID DisplayFamily_DisplayModel	Location
	06_3EH	See Table 2-26
	06_3F	See Table 2-32
	06_56H, 06_4FH	See Table 2-37
	06_4FH	See Table 2-38
IA32_MC1	9_CTL / MSR_MC19_CTL	
	06_2EH	See Table 2-17
	06_2DH	See Table 2-23
	06_3EH	
	06_3F	See Table 2-32
	06_56H, 06_4FH	See Table 2-37
	06_4FH	
IA32 MC1	9_MISC / MSR_MC19_MISC	
_	06_2EH	See Table 2-17
	06_3F	
	06_56H, 06_4FH	
	06_4FH	
IA32 MC1	9_STATUS / MSR_MC19_STATUS	000 100.0 0 00
	06_2EH	See Table 2-17
	06_2DH	
	06_3EH	
	06_3F	
	06_56H, 06_4FH	
	06_4FH	
1Δ32 MC2	_MISC / MSR_MC2_MISC	See Tuble 2 Se
1/\3Z_1 \CZ	06_1AH, 06_1EH, 06_1FH, 06_2EH	Soo Table 2-15
1Δ32 MC2	0_ADDR / MSR_MC20_ADDR	See Tuble 2 13
1/13Z_1 ICZ	06_2EH	Soo Table 2-17
	06_3EH	
	06_3F	
IVOS WCS	06_4FH 0_CTL / MSR_MC20_CTL	See Table 2-30
IA3Z_MCZ	06_2EH	Soo Table 2 17
	_	
	06_3EH	
	06_3F	
IVOS MCS		266 Janie 7-20
ıMJZ_MLZ	0_MISC / MSR_MC20_MISC	Coo Table 2 17
	06_2EH	
	06_3EH	
	06_3F	See Table 2-32
	UU 4FA	266 19(1)(6 V-2)

MSR Name and CPUID DisplayFamily_DisplayModel	Location
IA32_MC20_STATUS / MSR_MC20_STATUS	
06_2EH	See Table 2-17
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC21_ADDR / MSR_MC21_ADDR	
06_2EH	See Table 2-17
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4F	See Table 2-38
IA32_MC21_CTL / MSR_MC21_CTL	
06_2EH	See Table 2-17
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4F	See Table 2-38
IA32_MC21_MISC / MSR_MC21_MISC	
06_2EH	See Table 2-17
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4F	See Table 2-38
IA32_MC21_STATUS / MSR_MC21_STATUS	
06_2EH	See Table 2-17
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4F	See Table 2-38
IA32_MC22_ADDR / MSR_MC22_ADDR	
06_3EH	See Table 2-26
IA32_MC22_CTL / MSR_MC22_CTL	
06_3EH	See Table 2-26
IA32_MC22_MISC / MSR_MC22_MISC	
06_3EH	See Table 2-26
IA32_MC22_STATUS / MSR_MC22_STATUS	
06_3EH	See Table 2-26
IA32_MC23_ADDR / MSR_MC23_ADDR	
06_3EH	See Table 2-26
IA32_MC23_CTL / MSR_MC23_CTL	
06_3EH	See Table 2-26
IA32_MC23_MISC / MSR_MC23_MISC	
06_3EH	See Table 2-26
IA32_MC23_STATUS / MSR_MC23_STATUS	
06_3EH	See Table 2-26
IA32_MC24_ADDR / MSR_MC24_ADDR	

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_3EH	See Table 2-26
IA32_MC24_CTL / MSR_MC24_CTL	
06_3EH	See Table 2-26
IA32_MC24_MISC / MSR_MC24_MISC	
06_3EH	See Table 2-26
IA32_MC24_STATUS / MSR_MC24_STATUS	
06_3EH	See Table 2-26
IA32_MC25_ADDR / MSR_MC25_ADDR	
06_3EH	See Table 2-26
IA32_MC25_CTL / MSR_MC25_CTL	
06_3EH	See Table 2-26
IA32_MC25_MISC / MSR_MC25_MISC	
06_3EH	See Table 2-26
IA32_MC25_STATUS / MSR_MC25_STATUS	
06_3EH	See Table 2-26
IA32_MC26_ADDR / MSR_MC26_ADDR	
06_3EH	See Table 2-26
IA32_MC26_CTL / MSR_MC26_CTL	
06_3EH	See Table 2-26
IA32_MC26_MISC / MSR_MC26_MISC	
06_3EH	See Table 2-26
IA32_MC26_STATUS / MSR_MC26_STATUS	
06_3EH	See Table 2-26
IA32_MC27_ADDR / MSR_MC27_ADDR	
06_3EH	See Table 2-26
IA32_MC27_CTL / MSR_MC27_CTL	
06_3EH	See Table 2-26
IA32_MC27_MISC / MSR_MC27_MISC	
06_3EH	See Table 2-26
IA32_MC27_STATUS / MSR_MC27_STATUS	
06_3EH	See Table 2-26
IA32_MC28_ADDR / MSR_MC28_ADDR	
06_3EH	See Table 2-26
IA32_MC28_CTL / MSR_MC28_CTL	
06_3EH	See Table 2-26
IA32_MC28_MISC / MSR_MC28_MISC	
06_3EH	See Table 2-26
IA32_MC28_STATUS / MSR_MC28_STATUS	
06_3EH	See Table 2-26
IA32_MC29_ADDR / MSR_MC29_ADDR	
06_3EH	See Table 2-27
IA32_MC29_CTL / MSR_MC29_CTL	

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_3EH	See Table 2-27
IA32_MC29_MISC / MSR_MC29_MISC	
06_3EH	See Table 2-27
IA32_MC29_STATUS / MSR_MC29_STATUS	
06_3EH	See Table 2-27
IA32_MC3_ADDR / MSR_MC3_ADDR	
06_0FH, 06_17H	See Table 2-3
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_57H, 06_85H	See Table 2-53
06_0EH	See Table 2-58
06_09H	See Table 2-59
IA32_MC3_CTL / MSR_MC3_CTL	
06_0FH, 06_17H	See Table 2-3
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_57H, 06_85H	See Table 2-53
06_0EH	See Table 2-58
06_09H	See Table 2-59
IA32_MC3_MISC / MSR_MC3_MISC	
06_0FH, 06_17H	See Table 2-3
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_0EH	See Table 2-58
IA32_MC3_STATUS / MSR_MC3_STATUS	
06_0FH, 06_17H	See Table 2-3
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_57H, 06_85H	See Table 2-53
06_0EH	See Table 2-58
06_09H	See Table 2-59
IA32_MC30_ADDR / MSR_MC30_ADDR	
06_3EH	See Table 2-27
IA32_MC30_CTL / MSR_MC30_CTL	
06_3EH	See Table 2-27
IA32_MC30_MISC / MSR_MC30_MISC	
 06_3EH	See Table 2-27
IA32_MC30_STATUS / MSR_MC30_STATUS	
 06_3EH	See Table 2-27
IA32 MC31 ADDR / MSR MC31 ADDR	

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_3EH	See Table 2-27
IA32_MC31_CTL / MSR_MC31_CTL	
06_3EH	See Table 2-27
IA32_MC31_MISC / MSR_MC31_MISC	
06_3EH	See Table 2-27
IA32_MC31_STATUS / MSR_MC31_STATUS	
06_3EH	See Table 2-27
IA32_MC4_ADDR / MSR_MC4_ADDR	
06_0FH, 06_17H	See Table 2-3
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_57H, 06_85H	See Table 2-53
06_0EH	See Table 2-58
06_09H	See Table 2-59
IA32_MC4_CTL / MSR_MC4_CTL	
06_0FH, 06_17H	See Table 2-3
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_57H, 06_85H	See Table 2-53
06_0EH	See Table 2-58
06_09H	See Table 2-59
IA32_MC4_CTL2 / MSR_MC4_CTL2	
06_2AH, 06_2DH	See Table 2-20
IA32_MC4_STATUS / MSR_MC4_STATUS	
06_0FH, 06_17H	See Table 2-3
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_57H, 06_85H	See Table 2-53
06_0EH	See Table 2-58
06_09H	See Table 2-59
MSR_MC5_ADDR / MSR_MC5_ADDR	
06_0FH, 06_17H	See Table 2-3
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3FH	See Table 2-32
06_4FH	See Table 2-38
06_57H, 06_85H	See Table 2-53

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_0EH	See Table 2-58
IA32_MC5_CTL / MSR_MC5_CTL	
06_0FH, 06_17H	See Table 2-3
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3FH	See Table 2-32
06_4FH	See Table 2-38
06_57H, 06_85H	See Table 2-53
06_0EH	See Table 2-58
IA32_MC5_MISC / MSR_MC5_MISC	
06_0FH, 06_17H	See Table 2-3
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3FH	See Table 2-32
06_4FH	See Table 2-38
06_0EH	See Table 2-58
IA32_MC5_STATUS / MSR_MC5_STATUS	
06_0FH, 06_17H	See Table 2-3
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3FH	See Table 2-32
06_4FH	See Table 2-38
06_57H, 06_85H	See Table 2-53
06_0EH	See Table 2-58
IA32_MC6_ADDR / MSR_MC6_ADDR	
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38
IA32_MC6_CTL / MSR_MC6_CTL	
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_4FH	See Table 2-38
MSR_MC6_DEMOTION_POLICY_CONFIG	
06_37H	See Table 2-9
IA32_MC6_MISC / MSR_MC6_MISC	
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38
MSR_MC6_RESIDENCY_COUNTER	
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
06_37H	See Table 2-9
06_57H, 06_85H	See Table 2-53
IA32_CORE_CAPABILITIES (Note there are no architecturally defined bits; all bits are model-specific)	
06_86H	See Table 2-14
06_8CH, 06_8DH	See Table 2-45
IA32_MC6_STATUS / MSR_MC6_STATUS	
06_0FH, 06_17H	See Table 2-3
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3FH	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38
IA32_MC7_ADDR / MSR_MC7_ADDR	
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38
IA32_MC7_CTL / MSR_MC7_CTL	
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38
IA32_MC7_MISC / MSR_MC7_MISC	
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2DH	See Table 2-23

MODEL-SPECIFIC REGISTERS (MSRS)

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38
IA32_MC7_STATUS / MSR_MC7_STATUS	
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38
IA32_MC8_ADDR / MSR_MC8_ADDR	
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC8_CTL / MSR_MC8_CTL	
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC8_MISC / MSR_MC8_MISC	
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC8_STATUS / MSR_MC8_STATUS	
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_4FH	See Table 2-38
IA32_MC9_ADDR / MSR_MC9_ADDR	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38

MSR Name and CPUID DisplayFamily_DisplayModel	Location
IA32_MC9_CTL / MSR_MC9_CTL	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38
IA32_MC9_MISC / MSR_MC9_MISC	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38
IA32_MC9_STATUS / MSR_MC9_STATUS	
06_2EH	See Table 2-17
06_2DH	See Table 2-23
06_3EH	See Table 2-26
06_3F	See Table 2-32
06_56H, 06_4FH	See Table 2-37
06_4FH	See Table 2-38
MSR_MCG_MISC	
0FH	See Table 2-55
MSR_MCG_R10	
0FH	See Table 2-55
MSR_MCG_R11	
0FH	See Table 2-55
MSR_MCG_R12	
0FH	See Table 2-55
MSR_MCG_R13	
0FH	See Table 2-55
MSR_MCG_R14	
0FH	See Table 2-55
MSR_MCG_R15	
0FH	See Table 2-55
MSR_MCG_R8	
0FH	See Table 2-55
MSR_MCG_R9	
0FH	See Table 2-55
MSR_MCG_RAX	
0FH	See Table 2-55
MSR_MCG_RBP	

MODEL-SPECIFIC REGISTERS (MSRS)

MSR Name and CPUID DisplayFamily_DisplayModel	Location
0FH	See Table 2-55
MSR_MCG_RBX	
0FH	See Table 2-55
MSR_MCG_RCX	
0FH	See Table 2-55
MSR_MCG_RDI	
0FH	See Table 2-55
MSR_MCG_RDX	
0FH	See Table 2-55
MSR_MCG_RESERVED1 - MSR_MCG_RESERVED5	
0FH	See Table 2-55
MSR_MCG_RFLAGS	
0FH	See Table 2-55
MSR_MCG_RIP	
0FH	See Table 2-55
MSR_MCG_RSI	
0FH	See Table 2-55
MSR_MCG_RSP	
0FH	See Table 2-55
MSR_MEMORY_CTRL	
06_86H	See Table 2-14
06_7DH, 06_7EH	See Table 2-44
06_97H, 06_9AH	See Table 2-46
MSR_MISC_FEATURE_CONTROL	
06_5CH, 06_7AH	See Table 2-12
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	See Table 2-20
MSR_MISC_PWR_MGMT	
06_5CH, 06_7AH	See Table 2-12
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	See Table 2-20
MSR_MOB_ESCR0	
0FH	See Table 2-55
MSR_MOB_ESCR1	
0FH	See Table 2-55
MSR_MS_CCCR0	
0FH	See Table 2-55
MSR_MS_CCCR1	
0FH	See Table 2-55
MSR_MS_CCCR2	
0FH	See Table 2-55
MSR_MS_CCCR3	

MSR Name and CPUID DisplayFamily_DisplayModel	Location
0FH	See Table 2-55
MSR_MS_COUNTER0	
0FH	See Table 2-55
MSR_MS_COUNTER1	
0FH	See Table 2-55
MSR_MS_COUNTER2	
0FH	See Table 2-55
MSR_MS_COUNTER3	
0FH	See Table 2-55
MSR_MS_ESCR0	
0FH	See Table 2-55
MSR_MS_ESCR1	
0FH	See Table 2-55
MSR_MTRRCAP	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH	See Table 2-39
MSR_OFFCORE_RSP_0	
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	See Table 2-20
06_57H, 06_85H	See Table 2-53
MSR_OFFCORE_RSP_1	
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
06_25H, 06_2CH	See Table 2-18
06_2FH	See Table 2-19
06_2AH, 06_2DH	See Table 2-20
06_57H, 06_85H	See Table 2-53
MSR_PACKAGE_ENERGY_TIME_STATUS	
06_6AH, 06_6CH	See Table 2-51
MSR_PCIE_PLL_RATIO	
06_3FH	See Table 2-32
MSR_PCU_PMON_BOX_CTL	
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_PCU_PMON_BOX_FILTER	
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_PCU_PMON_BOX_STATUS	
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_PCU_PMON_CTR0	
06_2DH	See Table 2-24
06_3FH	See Table 2-33

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_PCU_PMON_CTR1	
06_2DH	. See Table 2-24
06_3FH	. See Table 2-33
MSR_PCU_PMON_CTR2	
06_2DH	. See Table 2-24
06_3FH	. See Table 2-33
MSR_PCU_PMON_CTR3	
06_2DH	. See Table 2-24
06_3FH	. See Table 2-33
MSR_PCU_PMON_EVNTSEL0	
06_2DH	. See Table 2-24
06_3FH	. See Table 2-33
MSR_PCU_PMON_EVNTSEL1	
06_2DH	. See Table 2-24
06_3FH	. See Table 2-33
MSR_PCU_PMON_EVNTSEL2	
06_2DH	. See Table 2-24
06_3FH	. See Table 2-33
MSR_PCU_PMON_EVNTSEL3	
06_2DH	. See Table 2-24
06_3FH	. See Table 2-33
MSR_PEBS_DATA_CFG	
06_7DH, 06_7EH	See Table 2-44
MSR_PEBS_ENABLE	
06_0FH, 06_17H	. See Table 2-3
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	. See Table 2-4
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	. See Table 2-7
06_5CH	. See Table 2-12
06_7AH	. See Table 2-13
06_86H	See Table 2-14
06_1AH, 06_1EH, 06_1FH, 06_2EH	. See Table 2-15
06_2AH, 06_2DH	. See Table 2-20
06_3EH	. See Table 2-27
06_57H, 06_85H	. See Table 2-53
0FH	See Table 2-55
MSR_PEBS_FRONTEND	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH	See Table 2-39
MSR_PEBS_LD_LAT	
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	. See Table 2-20
MSR_PEBS_MATRIX_VERT	
0FH	See Table 2-55

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_PEBS_NUM_ALT	
06_2DH	See Table 2-23
MSR_PERF_CAPABILITIES	
06_0FH, 06_17H	See Table 2-3
MSR_PERF_GLOBAL_CTRL	
06_0FH, 06_17H	See Table 2-3
MSR_PERF_GLOBAL_OVF_CTRL	
06_0FH, 06_17H	See Table 2-3
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
MSR_PERF_GLOBAL_STATUS	
06_0FH, 06_17H	See Table 2-3
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
MSR_PERF_METRICS	
06_7DH, 06_7EH	See Table 2-44
MSR_PERF_STATUS	
06_0FH, 06_17H	See Table 2-3
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
06_2AH, 06_2DH	See Table 2-20
MSR_PKG_C10_RESIDENCY	
06_5CH, 06_7AH	See Table 2-12
06_45H	See Table 2-30 and Table 2-31
06_4FH	See Table 2-38
MSR_PKG_C2_RESIDENCY	
06_27H	See Table 2-5
06_5CH, 06_7AH	See Table 2-12
06_2AH, 06_2DH, 06_3AH, 06_3CH, 06_3EH, 06_3FH, 06_45H, 06_46H	See Table 2-20
06_57H, 06_85H	See Table 2-53
MSR_PKG_C3_RESIDENCY	
06_5CH, 06_7AH	See Table 2-12
06_1AH, 06_1EH, 06_1FH, 06_2EH, 06_25H, 06_2CH, 06_2FH	See Table 2-15
06_2AH, 06_2DH, 06_3AH, 06_3CH, 06_3EH, 06_3FH, 06_45H, 06_46H	See Table 2-20
06_66H	See Table 2-42
06_57H, 06_85H	See Table 2-53
MSR_PKG_C4_RESIDENCY	
06_27H	See Table 2-5
MSR_PKG_C6_RESIDENCY	
06_27H	See Table 2-5
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
06_5CH, 06_7AH	See Table 2-12
06_1AH, 06_1EH, 06_1FH, 06_2EH, 06_25H, 06_2CH, 06_2FH	See Table 2-15
06 2AH, 06 2DH, 06 3AH, 06 3CH, 06 3EH, 06 3FH, 06 45H, 06 46H	See Table 2-20

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_57H, 06_85H	See Table 2-53
MSR_PKG_C7_RESIDENCY	
06_1AH, 06_1EH, 06_1FH, 06_2EH, 06_25H, 06_2CH, 06_2FH	See Table 2-15
06_2AH, 06_2DH, 06_3AH, 06_3CH, 06_3EH, 06_3FH, 06_45H, 06_46H	See Table 2-20
06_57H, 06_85H	See Table 2-53
MSR_PKG_C8_RESIDENCY	
06_45H	See Table 2-31
06_4FH	See Table 2-38
MSR_PKG_C9_RESIDENCY	
06_45H	See Table 2-31
06_4FH	See Table 2-38
MSR_PKG_CST_CONFIG_CONTROL	
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	See Table 2-7
06_4CH	See Table 2-11
06_5CH, 06_7AH	See Table 2-12
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	See Table 2-20
06_3AH	See Table 2-25
06_3EH	See Table 2-26
06_3CH, 06_45H, 06_46H	See Table 2-30
06_45H	See Table 2-31
06_3F	See Table 2-32
06_3DH	See Table 2-35
06_56H, 06_4FH	See Table 2-36
06_57H, 06_85H	See Table 2-53
MSR_PKG_ENERGY_STATUS	
06_37H, 06_4AH, 06_4CH, 06_5AH, 06_5DH	See Table 2-8
06_5CH, 06_7AH, 06_86H	See Table 2-12
06_2AH, 06_2DH, 06_3AH, 06_3CH, 06_3DH, 06_3EH, 06_3FH, 06_45H, 06_46H, 06_47H, 06_4EH, 06_4FH, 06_55H, 06_56H, 06_5EH, 06_66H, 06_8EH, 06_9EH, 06_7DH, 06_7EH	See Table 2-20
06_57H, 06_85H	See Table 2-53
MSR_PKG_HDC_CONFIG	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_PKG_HDC_DEEP_RESIDENCY	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_PKG_HDC_SHALLOW_RESIDENCY	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_PKG_PERF_STATUS	
06_5CH, 06_7AH, 06_86H	See Table 2-12
06_2DH	See Table 2-23

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_3AH, 06_3EH	. See Table 2-26
06_3CH, 06_3DH, 06_3FH, 06_45H, 06_46H, 06_47H, 06_4EH, 06_4FH, 06_55H,	See Table 2-29
06_56H, 06_5EH, 06_66H, 06_8EH, 06_9EH, 06_7DH, 06_7EH	
06_57H, 06_85H	. See Table 2-53
MSR_PKG_POWER_INFO	
06_4DH	. See Table 2-10
06_5CH, 06_7AH, 06_86H	. See Table 2-12
06_2AH, 06_2DH, 06_3AH, 06_3DH, 06_3CH, 06_3EH, 06_3FH, 06_45H, 06_46H, 06_47H, 06_4EH, 06_4EH, 06_4EH, 06_5EH, 06_5EH, 06_6EH, 06_8EH, 06_9EH, 06_7DH, 06_7EH	See Table 2-20
06_57H, 06_85H	. See Table 2-53
MSR_PKG_POWER_LIMIT	
06_37H, 06_4AH, 06_4CH, 06_5AH, 06_5DH	. See Table 2-8
06_4DH	. See Table 2-10
06_5CH, 06_7AH, 06_86H	
06_2AH, 06_2DH, 06_3AH, 06_3CH, 06_3DH, 06_3EH, 06_3FH, 06_45H, 06_46H, 06_47H, 06_4EH, 06_4FH, 06_55H, 06_5EH, 06_6EH, 06_8EH, 06_9EH, 06_7DH, 06_7EH	See Table 2-20
06_57H, 06_85H	. See Table 2-53
MSR_PKGC_IRTL1	
06_5CH, 06_7AH	. See Table 2-12
06_3CH, 06_45H, 06_46H	. See Table 2-29
MSR_PKGC_IRTL2	
06_5CH, 06_7AH	. See Table 2-12
06_3CH, 06_45H, 06_46H	
MSR_PKGC3_IRTL	
 06_5CH, 06_7AH	. See Table 2-12
_	
MSR_PKGC6_IRTL	
06_2AH, 06_2DH	. See Table 2-20
MSR_PKGC7_IRTL	
06_2AH	See Table 2-21
MSR_PLATFORM_BRV	. 500 10510 2 2 1
0FH	See Table 2-55
MSR_PLATFORM_ENERGY_COUNTER	See Tuble 2 33
06_4EH, 06_5EH, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH,	See Table 2-39
06_6AH, 06_6CH	
MSR_PLATFORM_ID	
06_0FH, 06_17H	. See Table 2-3
06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH	
06_5CH, 06_7AH	
06_1AH, 06_1EH, 06_1FH, 06_2EH	
MSR_PLATFORM_INFO	300 . 30.0 2 13
06_5CH, 06_7AH	. See Table 2-12

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	See Table 2-20
06_3AH	See Table 2-25
06_3EH	See Table 2-26
06_3CH, 06_45H, 06_46H	See Table 2-29 and Table 2-30
06_56H, 06_4FH	See Table 2-36
06_57H	See Table 2-53
MSR_PLATFORM_POWER_LIMIT	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MSR_PMG_IO_CAPTURE_BASE	
06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
06_4CH	See Table 2-11
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	See Table 2-20
06_3AH	See Table 2-25
06_3EH	See Table 2-26
06_57H	See Table 2-53
MSR_PMH_ESCR0	
0FH	See Table 2-55
MSR_PMH_ESCR1	
0FH	See Table 2-55
MSR_PMON_GLOBAL_CONFIG	
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_PMON_GLOBAL_CTL	
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_PMON_GLOBAL_STATUS	
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_POWER_CTL	
06_5CH, 06_7AH	
06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06_2AH, 06_2DH	See Table 2-20
MSR_PPO_ENERGY_STATUS	
06_37H, 06_4AH, 06_5AH, 06_5DH	See Table 2-8
06_5CH, 06_7AH	See Table 2-12
06_2AH, 06_2DH, 06_3AH, 06_3CH, 06_3EH, 06_3FH, 06_45H, 06_46H	See Table 2-20
06_57H	See Table 2-53
MSR PPO POLICY	

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_2AH, 06_45H	See Table 2-21
MSR_PPO_POWER_LIMIT	
06_4CH	See Table 2-11
06_2AH, 06_2DH, 06_3AH, 06_3CH, 06_3EH, 06_3FH, 06_45H, 06_46H	See Table 2-20
06_57H	See Table 2-53
MSR_PP1_ENERGY_STATUS	
06_5CH, 06_7AH	See Table 2-12
06_2AH, 06_45H	See Table 2-21
06_3CH, 06_45H, 06_46H	See Table 2-30
MSR_PP1_POLICY	
06_2AH, 06_45H	See Table 2-21
06_3CH, 06_45H, 06_46H	See Table 2-30
MSR_PP1_POWER_LIMIT	
06_2AH, 06_45H	See Table 2-21
06_3CH, 06_45H, 06_46H	See Table 2-30
MSR_PPERF	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
IA32_PPIN / MSR_PPIN	
06_3EH	See Table 2-26
06_56H, 06_4FH	See Table 2-36
06_55H	See Table 2-50
06_57H, 06_85H	See Table 2-53
IA32_PPIN_CTL / MSR_PPIN_CTL	
06_3EH	See Table 2-26
06_56H, 06_4FH	See Table 2-36
06_55H	See Table 2-50
06_57H, 06_85H	See Table 2-53
MSR_PRMRR_BASE_0	
06_7DH, 06_7EH	See Table 2-44
MSR_PRMRR_PHYS_BASE	
06_8EH, 06_9EH	See Table 2-41
MSR_PRMRR_PHYS_MASK	
06_8EH, 06_9EH	See Table 2-41
MSR_PRMRR_VALID_CONFIG	
06_8EH, 06_9EH	See Table 2-41
MSR_RELOAD_FIXED_CTRx	
06_86H	See Table 2-14
MSR_RELOAD_PMCx	
06_86H	See Table 2-14
MSR_RING_RATIO_LIMIT	
06 8EH, 06 9EH	See Table 2-41

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_R0_PMON_BOX_CTRL	
06_2EH	See Table 2-17
MSR_R0_PMON_BOX_OVF_CTRL	
06_2EH	See Table 2-17
MSR_R0_PMON_BOX_STATUS	
06_2EH	See Table 2-17
MSR_R0_PMON_CTR0	
06_2EH	See Table 2-17
MSR_R0_PMON_CTR1	
06_2EH	See Table 2-17
MSR_R0_PMON_CTR2	
06_2EH	See Table 2-17
MSR_R0_PMON_CTR3	
06_2EH	See Table 2-17
MSR_R0_PMON_CTR4	
06_2EH	See Table 2-17
MSR_R0_PMON_CTR5	
06_2EH	See Table 2-17
MSR_R0_PMON_CTR6	
06_2EH	See Table 2-17
MSR_R0_PMON_CTR7	
06_2EH	See Table 2-17
MSR_R0_PMON_EVNT_SEL0	
06_2EH	See Table 2-17
MSR_R0_PMON_EVNT_SEL1	
06_2EH	See Table 2-17
MSR_R0_PMON_EVNT_SEL2	
06_2EH	See Table 2-17
MSR_R0_PMON_EVNT_SEL3	
06_2EH	See Table 2-17
MSR_R0_PMON_EVNT_SEL4	
06_2EH	See Table 2-17
MSR_R0_PMON_EVNT_SEL5	
06_2EH	See Table 2-17
MSR_R0_PMON_EVNT_SEL6	
06_2EH	See Table 2-17
MSR_R0_PMON_EVNT_SEL7	
06_2EH	See Table 2-17
MSR_RO_PMON_IPERFO_PO	
06_2EH	See Table 2-17
MSR_R0_PMON_IPERF0_P1	
06_2EH	See Table 2-17

MSR Name and CPUID DisplayFamily	<i>y_</i> DisplayModel	Location
MSR_R0_PMON_IPERF0_P2		
06_2EH		See Table 2-17
MSR_R0_PMON_IPERF0_P3		
06_2EH		See Table 2-17
MSR_R0_PMON_IPERF0_P4		
06_2EH		See Table 2-17
MSR_R0_PMON_IPERF0_P5		
06_2EH		See Table 2-17
MSR_R0_PMON_IPERF0_P6		
06_2EH		See Table 2-17
MSR_R0_PMON_IPERF0_P7		
06_2EH		See Table 2-17
MSR_R0_PMON_QLX_P0		
06_2EH		See Table 2-17
MSR_R0_PMON_QLX_P1		
06_2EH		See Table 2-17
MSR_R0_PMON_QLX_P2		
06_2EH		See Table 2-17
MSR_R0_PMON_QLX_P3		
06_2EH		See Table 2-17
MSR_R1_PMON_BOX_CTRL		
06_2EH		See Table 2-17
MSR_R1_PMON_BOX_OVF_CTRL		
06_2EH		See Table 2-17
MSR_R1_PMON_BOX_STATUS		
06_2EH		See Table 2-17
MSR_R1_PMON_CTR10		
06_2EH		See Table 2-17
MSR_R1_PMON_CTR11		
06_2EH		See Table 2-17
MSR_R1_PMON_CTR12		
06_2EH		See Table 2-17
MSR_R1_PMON_CTR13		
_		See Table 2-17
MSR_R1_PMON_CTR14		
		See Table 2-17
MSR_R1_PMON_CTR15		
		See Table 2-17
MSR_R1_PMON_CTR8		
		See Table 2-17
MSR_R1_PMON_CTR9		
06_2EH		See Table 2-17

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_R1_PMON_EVNT_SEL10	
06_2EH	See Table 2-17
MSR_R1_PMON_EVNT_SEL11	
06_2EH	. See Table 2-17
MSR_R1_PMON_EVNT_SEL12	
06_2EH	. See Table 2-17
MSR_R1_PMON_EVNT_SEL13	
06_2EH	. See Table 2-17
MSR_R1_PMON_EVNT_SEL14	
06_2EH	. See Table 2-17
MSR_R1_PMON_EVNT_SEL15	
06_2EH	. See Table 2-17
MSR_R1_PMON_EVNT_SEL8	
06_2EH	. See Table 2-17
MSR_R1_PMON_EVNT_SEL9	
06_2EH	. See Table 2-17
MSR_R1_PMON_IPERF1_P10	
06_2EH	. See Table 2-17
MSR_R1_PMON_IPERF1_P11	
06_2EH	See Table 2-17
MSR_R1_PMON_IPERF1_P12	
06_2EH	See Table 2-17
MSR_R1_PMON_IPERF1_P13	
06_2EH	. See Table 2-17
MSR_R1_PMON_IPERF1_P14	
06_2EH	. See Table 2-17
MSR_R1_PMON_IPERF1_P15	
06_2EH	. See Table 2-17
MSR_R1_PMON_IPERF1_P8	
06_2EH	See Table 2-17
MSR_R1_PMON_IPERF1_P9	
06_2EH	See Table 2-17
MSR_R1_PMON_QLX_P4	
06_2EH	See Table 2-17
MSR_R1_PMON_QLX_P5	
06_2EH	See Table 2-17
MSR_R1_PMON_QLX_P6	
06_2EH	See Table 2-17
MSR_R1_PMON_QLX_P7	
06_2EH	See Table 2-17
MSR_RAPL_POWER_UNIT	
06_37H, 06_4AH, 06_5AH, 06_5DH	See Table 2-8

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_4DH	See Table 2-10
06_5CH, 06_7AH	See Table 2-12
06_2AH, 06_2DH, 06_3AH, 06_3CH, 06_3EH, 06_3FH, 06_45H, 06_46H	See Table 2-20
06_3FH	See Table 2-32
06_56H, 06_4FH	See Table 2-36
06_57H	See Table 2-53
MSR_RAT_ESCRO	
0FH	See Table 2-55
MSR_RAT_ESCR1	
0FH	See Table 2-55
MSR_RING_PERF_LIMIT_REASONS	
06_3CH, 06_45H, 06_46H	See Table 2-30
MSR_SO_PMON_BOX_CTRL	
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_SO_PMON_BOX_FILTER	
06_3FH	See Table 2-33
MSR_SO_PMON_BOX_OVF_CTRL	
06_2EH	See Table 2-17
MSR_SO_PMON_BOX_STATUS	
06_2EH	See Table 2-17
MSR_SO_PMON_CTR0	
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_SO_PMON_CTR1	
06_2EH	
06_3FH	See Table 2-33
MSR_SO_PMON_CTR2	
06_2EH	
06_3FH	See Table 2-33
MSR_SO_PMON_CTR3	
06_2EH	
06_3FH	See Table 2-33
MSR_SO_PMON_EVNT_SEL0	
06_2EH	
06_3FH	See Table 2-33
MSR_SO_PMON_EVNT_SEL1	
06_2EH	
06_3FH	See Table 2-33
MSR_SO_PMON_EVNT_SEL2	
06_2EH	
06_3FH	See Table 2-33

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_SO_PMON_EVNT_SEL3	
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_SO_PMON_MASK	
06_2EH	See Table 2-17
MSR_SO_PMON_MATCH	
06_2EH	See Table 2-17
MSR_S1_PMON_BOX_CTRL	
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_S1_PMON_BOX_FILTER	
06_3FH	See Table 2-33
MSR_S1_PMON_BOX_OVF_CTRL	
06_2EH	See Table 2-17
MSR_S1_PMON_BOX_STATUS	
06_2EH	See Table 2-17
MSR_S1_PMON_CTR0	
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_S1_PMON_CTR1	
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_S1_PMON_CTR2	
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_S1_PMON_CTR3	
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_S1_PMON_EVNT_SEL0	
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_S1_PMON_EVNT_SEL1	
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_S1_PMON_EVNT_SEL2	
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_S1_PMON_EVNT_SEL3	
06_2EH	See Table 2-17
06_3FH	See Table 2-33
MSR_S1_PMON_MASK	
06_2EH	See Table 2-17

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_S1_PMON_MATCH	
06_2EH	See Table 2-17
MSR_S2_PMON_BOX_CTL	
06_3FH	See Table 2-33
MSR_S2_PMON_BOX_FILTER	
06_3FH	See Table 2-33
MSR_S2_PMON_CTR0	
06_3FH	See Table 2-33
MSR_S2_PMON_CTR1	
06_3FH	See Table 2-33
MSR_S2_PMON_CTR2	
06_3FH	See Table 2-33
MSR_S2_PMON_CTR3	
06_3FH	See Table 2-33
MSR_S2_PMON_EVNTSEL0	
06_3FH	See Table 2-33
MSR_S2_PMON_EVNTSEL1	
06_3FH	See Table 2-33
MSR_S2_PMON_EVNTSEL2	
06_3FH	See Table 2-33
MSR_S2_PMON_EVNTSEL3	
06_3FH	See Table 2-33
MSR_S3_PMON_BOX_CTL	
06_3FH	See Table 2-33
MSR_S3_PMON_BOX_FILTER	
06_3FH	See Table 2-33
MSR_S3_PMON_CTR0	
06_3FH	See Table 2-33
MSR_S3_PMON_CTR1	
06_3FH	See Table 2-33
MSR_S3_PMON_CTR2	
06_3FH	See Table 2-33
MSR_S3_PMON_CTR3	
06_3FH	See Table 2-33
MSR_S3_PMON_EVNTSEL0	
06_3FH	See Table 2-33
MSR_S3_PMON_EVNTSEL1	
06_3FH	See Table 2-33
MSR_S3_PMON_EVNTSEL2	
06_3FH	See Table 2-33
MSR_S3_PMON_EVNTSEL3	
06_3FH	See Table 2-33

MSR Name an	nd CPUID DisplayFamily_DisplayModel	Location
MSR_SAAT_ES	SCRO	
0F	H	See Table 2-55
MSR_SAAT_ES	SCR1	
0F	H	See Table 2-55
MSR_SGXOWN	NEREPOCHO	
06	S_5CH, 06_7AH	See Table 2-12
	6_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 6_6AH, 06_6CH	See Table 2-39
MSR_SGXOWN	NEREPOCH1	
06	S_5CH, 06_7AH	See Table 2-12
	6_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 6_6AH, 06_6CH	See Table 2-39
MSR_SMI_COU	JNT	
06	5_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
06	5_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
06	5_2AH, 06_2DH	See Table 2-20
06	5_57H	See Table 2-53
MSR_SMM_BL	OCKED	
06	S_5CH, 06_7AH	See Table 2-12
06	S_3CH, 06_45H, 06_46H	See Table 2-30
MSR_SMM_DE	ELAYED	
06	S_5CH, 06_7AH	See Table 2-12
06	S_3CH, 06_45H, 06_46H	See Table 2-30
MSR_SMM_FE	ATURE_CONTROL	
06	S_5CH, 06_7AH	See Table 2-12
06	S_3CH, 06_45H, 06_46H	See Table 2-30
MSR_SMM_MC	CA_CAP	
06	S_5CH, 06_7AH	See Table 2-12
06	5_3CH, 06_45H, 06_46H	See Table 2-30
06	5_3FH	See Table 2-32
06	5_56H, 06_4FH	See Table 2-36
06	5_57H	See Table 2-53
MSR_SMRR_P	HYSBASE	
06	5_0FH, 06_17H	See Table 2-3
MSR_SMRR_P	HYSMASK	
06	5_0FH, 06_17H	See Table 2-3
MSR_SSU_ESC	CRO	
0F	H	See Table 2-55
MSR_TBPU_ES	SCR0	
0F	TH	See Table 2-55
MSR_TBPU_ES	SCR1	
0F	Ή	See Table 2-55

MSR Na	me and CPUID DisplayFamily_DisplayModel	Location
MSR_TC	_ESCRO	
	0FH	See Table 2-55
MSR_TC	_ESCR1	
	0FH	See Table 2-55
MSR_TC	_PRECISE_EVENT	
	0FH	See Table 2-55
MSR_TE	MPERATURE_TARGET	
	06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
	06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
	06_2AH, 06_2DH	See Table 2-20
	06_3EH	See Table 2-26
	06_56H, 06_4FH	See Table 2-36
	06_57H	See Table 2-53
MSR_TE	ST_CTRL	
	P6 Family	See Table 2-60
MSR_TH	HERM2_CTL	
	06_0FH, 06_17H	See Table 2-3
	06_1CH, 06_26H, 06_27H, 06_35H, 06_36H	See Table 2-4
	0FH	See Table 2-55
	06_0EH	See Table 2-58
	06_09H	See Table 2-59
MSR_TH	HREAD_ID_INFO	
	06_3FH	See Table 2-32
MSR_TR	race_hub_sth_acpibar_base	
	06_8EH, 06_9EH	See Table 2-41
MSR_TU	Jrbo_activation_ratio	
	06_5CH, 06_7AH	See Table 2-12
	06_3AH	See Table 2-25
	06_3CH, 06_45H, 06_46H	See Table 2-29
	06_57H	See Table 2-53
MSR_TU	JRBO_GROUP_CORECNT	
	06_5CH, 06_7AH	See Table 2-12
MSR_TU	Jrbo_power_current_limit	
	06_1AH, 06_1EH, 06_1FH, 06_2EH	See Table 2-15
MSR_TU	JRBO_RATIO_LIMIT	
	06_37H, 06_4AH, 06_4DH, 06_5AH, 06_5DH, 06_5CH, 06_7AH	See Table 2-6
	06_4DH	See Table 2-10
	06_5CH, 06_7AH	See Table 2-12
	06_1AH, 06_1EH, 06_1FH, 06_2EH, 06_25H, 06_2CH	See Table 2-15
	06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
	06_2EH	See Table 2-17
	06_25H, 06_2CH	See Table 2-18

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_2FH	See Table 2-19
06_2AH, 06_45H	See Table 2-21
06_2DH	See Table 2-23
06_3EH	See Table 2-26 and Table 2-27
06_3CH, 06_45H, 06_46H	See Table 2-30
06_3FH	See Table 2-32
06_3DH	See Table 2-35
06_56H, 06_4FH	See Table 2-36
06_55H	See Table 2-50
06_57H	See Table 2-53
MSR_TURBO_RATIO_LIMIT1	
06_3EH	See Table 2-26 and Table 2-27
06_3FH	See Table 2-32
06_56H, 06_4FH	See Table 2-36
MSR_TURBO_RATIO_LIMIT2	
06_3FH	See Table 2-32
MSR_TURBO_RATIO_LIMIT3	
06_56H	See Table 2-37
06_4FH	See Table 2-38
MSR_TURBO_RATIO_LIMIT_CORES	
06_55H	See Table 2-50
MSR_U_PMON_BOX_STATUS	
06_3EH	See Table 2-28
06_3FH	See Table 2-33
MSR_U_PMON_CTR	
06_2EH	See Table 2-17
MSR_U_PMON_CTR0	
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_U_PMON_CTR1	
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_U_PMON_EVNT_SEL	
06_2EH	See Table 2-17
MSR_U_PMON_EVNTSEL0	
06_2DH	
06_3FH	See Table 2-33
MSR_U_PMON_EVNTSEL1	
06_2DH	See Table 2-24
06 3FH	See Table 2-33

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_U_PMON_GLOBAL_CTRL	
06_2EH	See Table 2-17
MSR_U_PMON_GLOBAL_OVF_CTRL	
06_2EH	See Table 2-17
MSR_U_PMON_GLOBAL_STATUS	
06_2EH	See Table 2-17
MSR_U_PMON_UCLK_FIXED_CTL	
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_U_PMON_UCLK_FIXED_CTR	
06_2DH	See Table 2-24
06_3FH	See Table 2-33
MSR_U2L_ESCR0	
0FH	See Table 2-55
MSR_U2L_ESCR1	
0FH	See Table 2-55
MSR_UNC_ARB_PERFCTRO	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNC_ARB_PERFCTR1	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNC_ARB_PERFEVTSEL0	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNC_ARB_PERFEVTSEL1	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNC_CBO_O_PERFCTRO	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNC_CBO_0_PERFCTR1	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNC_CBO_O_PERFCTR2	
06_2AH	See Table 2-22

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_UNC_CBO_0_PERFCTR3	
06_2AH	See Table 2-22
MSR_UNC_CBO_0_PERFEVTSEL0	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNC_CBO_0_PERFEVTSEL1	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNC_CBO_0_PERFEVTSEL2	
06_2AH	See Table 2-22
MSR_UNC_CBO_0_PERFEVTSEL3	
06_2AH	See Table 2-22
MSR_UNC_CBO_O_UNIT_STATUS	
06_2AH	See Table 2-22
MSR_UNC_CBO_1_PERFCTRO	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNC_CBO_1_PERFCTR1	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNC_CBO_1_PERFCTR2	
06_2AH	See Table 2-22
MSR_UNC_CBO_1_PERFCTR3	
06_2AH	See Table 2-22
MSR_UNC_CBO_1_PERFEVTSEL0	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNC_CBO_1_PERFEVTSEL1	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNC_CBO_1_PERFEVTSEL2	
06_2AH	See Table 2-22
MSR_UNC_CBO_1_PERFEVTSEL3	
06_2AH	See Table 2-22
MSR_UNC_CBO_1_UNIT_STATUS	
06_2AH	See Table 2-22

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_UNC_CBO_2_PERFCTRO	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNC_CBO_2_PERFCTR1	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNC_CBO_2_PERFCTR2	
06_2AH	See Table 2-22
MSR_UNC_CBO_2_PERFCTR3	
06_2AH	See Table 2-22
MSR_UNC_CBO_2_PERFEVTSELO	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNC_CBO_2_PERFEVTSEL1	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNC_CBO_2_PERFEVTSEL2	
06_2AH	See Table 2-22
MSR_UNC_CBO_2_PERFEVTSEL3	
06_2AH	See Table 2-22
MSR_UNC_CBO_2_UNIT_STATUS	
06_2AH	See Table 2-22
MSR_UNC_CBO_3_PERFCTRO	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNC_CBO_3_PERFCTR1	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNC_CBO_3_PERFCTR2	
06_2AH	See Table 2-22
MSR_UNC_CBO_3_PERFCTR3	
06_2AH	See Table 2-22
MSR_UNC_CBO_3_PERFEVTSELO	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_UNC_CBO_3_PERFEVTSEL1	
06_2AH	. See Table 2-22
06_3CH, 06_45H, 06_46H	. See Table 2-30
06_4EH, 06_5EH	. See Table 2-40
MSR_UNC_CBO_3_PERFEVTSEL2	
06_2AH	. See Table 2-22
MSR_UNC_CBO_3_PERFEVTSEL3	
06_2AH	. See Table 2-22
MSR_UNC_CBO_3_UNIT_STATUS	
06_2AH	. See Table 2-22
MSR_UNC_CBO_4_PERFCTRO	
06_2AH	. See Table 2-22
MSR_UNC_CBO_4_PERFCTR1	
06_2AH	. See Table 2-22
MSR_UNC_CBO_4_PERFCTR2	
06_2AH	. See Table 2-22
MSR_UNC_CBO_4_PERFCTR3	
06_2AH	. See Table 2-22
MSR_UNC_CBO_4_PERFEVTSELO	
06_2AH	. See Table 2-22
MSR_UNC_CBO_4_PERFEVTSEL1	
06_2AH	. See Table 2-22
MSR_UNC_CBO_4_PERFEVTSEL2	
06_2AH	. See Table 2-22
MSR_UNC_CBO_4_PERFEVTSEL3	
06_2AH	. See Table 2-22
MSR_UNC_CBO_4_UNIT_STATUS	
06_2AH	. See Table 2-22
MSR_UNC_CBO_CONFIG	
06_2AH	. See Table 2-22
06_3CH, 06_45H, 06_46H	. See Table 2-30
06_4EH, 06_5EH	. See Table 2-40
MSR_UNC_PERF_FIXED_CTR	
06_2AH	. See Table 2-22
06_3CH, 06_45H, 06_46H	. See Table 2-30
06_4EH, 06_5EH	. See Table 2-40
MSR_UNC_PERF_FIXED_CTRL	
06_2AH	. See Table 2-22
06_3CH, 06_45H, 06_46H	. See Table 2-30
06_4EH, 06_5EH	. See Table 2-40
MSR_UNC_PERF_GLOBAL_CTRL	
06_2AH	. See Table 2-22

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNC_PERF_GLOBAL_STATUS	
06_2AH	See Table 2-22
06_3CH, 06_45H, 06_46H	See Table 2-30
06_4EH, 06_5EH	See Table 2-40
MSR_UNCORE_ADDR_OPCODE_MATCH	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_FIXED_CTR_CTRL	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_FIXED_CTR0	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_PERF_GLOBAL_CTRL	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_PERF_GLOBAL_OVF_CTRL	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_PERF_GLOBAL_STATUS	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_PERFEVTSEL0	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_PERFEVTSEL1	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_PERFEVTSEL2	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_PERFEVTSEL3	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_PERFEVTSEL4	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_PERFEVTSEL5	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_PERFEVTSEL6	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_PERFEVTSEL7	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_PMCO	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_PMC1	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_PMC2	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_PMC3	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16

MSR Name and CPUID DisplayFamily_DisplayModel	Location
MSR_UNCORE_PMC4	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_PMC5	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
06_2EH	See Table 2-17
MSR_UNCORE_PMC6	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_PMC7	
06_1AH, 06_1EH, 06_1FH, 06_25H, 06_2CH	See Table 2-16
MSR_UNCORE_PRMRR_BASE	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH 06_6AH, 06_6CH	
MSR_UNCORE_PRMRR_MASK	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH 06_6AH, 06_6CH	
MSR_UNCORE_PRMRR_PHYS_BASE	
06_8EH, 06_9EH	See Table 2-41
MSR_UNCORE_PRMRR_PHYS_MASK	
06_8EH, 06_9EH	See Table 2-41
MSR_VR_CURRENT_CONFIG	
06_8CH, 06_8DH	See Table 2-45
MSR_W_PMON_BOX_CTRL	
06_2EH	See Table 2-17
MSR_W_PMON_BOX_OVF_CTRL	
06_2EH	See Table 2-17
MSR_W_PMON_BOX_STATUS	
06_2EH	See Table 2-17
MSR_W_PMON_CTR0	
06_2EH	See Table 2-17
MSR_W_PMON_CTR1	
06_2EH	See Table 2-17
MSR_W_PMON_CTR2	
06_2EH	See Table 2-17
MSR_W_PMON_CTR3	
06_2EH	See Table 2-17
MSR_W_PMON_EVNT_SELO	
06_2EH	See Table 2-17
MSR_W_PMON_EVNT_SEL1	
06_2EH	See Table 2-17
MSR_W_PMON_EVNT_SEL2	
06_2EH	See Table 2-17
MSR_W_PMON_EVNT_SEL3	

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_2EH	See Table 2-17
MSR_W_PMON_FIXED_CTR	
06_2EH	See Table 2-17
MSR_W_PMON_FIXED_CTR_CTL	
06_2EH	See Table 2-17
MSR_WEIGHTED_CORE_CO	
06_4EH, 06_5EH, 06_55H, 06_8EH, 06_9EH, 06_66H, 06_7DH, 06_7EH, 06_8CH, 06_8DH, 06_6AH, 06_6CH	See Table 2-39
MTRRfix16K_80000	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRfix16K_A0000	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRfix4K_C0000	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRfix4K_C8000	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRfix4K_D0000	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRfix4K_D8000	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRfix4K_E0000	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRfix4K_E8000	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRfix4K_F0000	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRfix4K_F8000	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRfix64K_00000	
06_0EH	
P6 Family	See Table 2-60
MTRRphysBase0	

MODEL-SPECIFIC REGISTERS (MSRS)

MSR Name and CPUID DisplayFamily_DisplayModel	Location
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRphysBase1	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRphysBase2	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRphysBase3	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRphysBase4	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRphysBase5	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRphysBase6	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRphysBase7	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRphysMask0	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRphysMask1	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRphysMask2	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRphysMask3	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRphysMask4	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRphysMask5	
06_0EH	See Table 2-58
P6 Family	See Table 2-60
MTRRphysMask6	

MSR Name	e and CPUID DisplayFamily_DisplayModel	Location
	06_0EH	See Table 2-58
	P6 Family	See Table 2-60
MTRRphysl	Mask7	
	06_0EH	See Table 2-58
	P6 Family	See Table 2-60

INDEX

Advanced programmable interrupt controller (see I/O APIC or Local APIC) Advanced programmable interrupt controller (see I/O APIC or Local APIC) B Backlink (see Previous task link) Backlink (see Previous task link) Bilt order, 1-5 Birt order, 1-5 Birt order, 1-5 Birt SullAVAIMABLE files, 1-8 Birt SullAVAIMABLE files, 2-379 Birt SullAVAIMABLE fi		
Addressing, segments, 1-6 APIC (see I/O APIC or Local APIC) B BACklink (see Previous task link) Binary numbers, 1-6 Binary numbers, 1-7 Binarch trace ressing (see BTM) Binary numbers, 1-8 Binary numbers,	Δ	IA32 MCi ADDR MSR. 2-388
Advanced programmable interrupt controller (see I/O APIC or Local APIC) B BACKIR (see I/O APIC or Local APIC) B Backlink (see Provious task link) Binary numbers, 1-8 Bit order, 1-5 B		
APIC (see I/O APIC or Local APIC) B Backlink (see Previous task link) Branch trace riscsage (see BTM) Branch trace riscsage (see BTM) Branch trace riscsage (see BTM) BTS (branch trace riscsage (see BTM) BTS (branch trace riscsage) BTS (branch trace) BTS (branch trace)		
Backlink (see Previous task link) Backlink (see Previous task link) Binary numbers, 1-6 Binary numbers, 1-6 Binary numbers, 1-6 Binary numbers, 1-7 Binary trace store (see BTM) Branch trace store (see BTS) Byte order, 1-5 Byte order,	Advanced programmable interrupt controller (see I/O APIC or Local APIC)	
Backlink (see Previous task link) Backlink (see Previous task link) Backlink (see Previous task link) Blit order, 1-5 Bit order, 1-5 Bit order, 1-5 Bit order, 1-5 Branch trace message (see BTM) Branch trace store (see BTS) See WMX capability MSRs See WMX c	APIC (see I/O APIC or Local APIC)	
Backlink (see Previous task link) Binary numbers, 1-6 Binary numbers, 1-6 Binary numbers, 1-7 Binary numbers, 1-8 Binary numbe		,
Backlink (see Previous task link) Binary numbers, 1-6 Bit order, 1-5 Bit order, 1		
Backlink (see Previous task link) Binary numbers, 1-6 Bit order, 1-5 Branch trace store (see BTM) Branch trace store (see BTM) Branch trace store (see BTM) Branch trace store) facilities Bit Spranch trace store) facilities See WIX capability MSRs See WIX capability MSRs See WIX capability MSRs Context, task (see Task state) CPUID instruction syntax for data, 1-7 Urrent privilege level (see CPL) D Debug store (see DS) Bebug s	В	IA32_MTRR_PHYS BASEn MTRR, 2-382
Binary numbers, 1-6	Backlink (see Previous task link)	
Bit order, 1-5		·
Branch trace message (see BTM)		= -
Branch trace store (see BTS)		
## BTS UNAVAILABLE flag ## BTS	Branch trace message (see BTM)	
## A32_MCC_RASE MSR_2-370 ## A32_MCC_RASE MSR_2-370 ## A32_MCC_RASE MSR_2-379 ## A32_MCC_RASE MSR_2-376 ## A32_MCC_RASE MSR_2-375 ## A32_MCC_RASE MS	Branch trace store (see BTS)	IA32_PEBS_ENABLE MSR, 2-387
## BYS UNAVAILABLE flag		IA32 PLATFORM ID. 2-67, 2-81, 2-141, 2-178, 2-353, 2-370, 2-396,
A32_MISC_ENABLE MSR, 2-379 Byte order, 1-5 A32_SYENTER_ EIP MSR, 2-373 A32_SYENTER_ EIP MSR, 2-374 A32_SYENTER_ EIP MSR, 2-373 A32_SYENTER_ EIP MSR, 2-377 A32_THEM_ INTERQUIT MSR, 2-377 A32_THEM_ INTERQUIT MSR, 2-377 A32_THEM_ INTERQUIT MSR, 2-370 A32_SYENTER_ EIP MSR, 2-370 A32_SYENTER_ EIP MSR, 2-376 A3		
A32_SYSENTER_CS MSR, 2-373 A32_SYSENTER_CS MSR, 2-373 A32_SYSENTER_ES MSR, 2-374 A32_SYSENTER_ES MSR, 2-375 A32_SYSENTER_ES MSR, 2-374 A32_SYSENTER_ES MSR, 2-375 A32_SYSENTER_ES MSR, 2-376 A32_SYSENTER_ES MSR, 2-375 A32_SYSENTER_ES MSR, 2-375 A32_SYSENTER_ES MSR, 2-376 A32_SYSENTER_S MSR, 2-375 A32_SYSENTER_S MSR, 2-375		
A32_SYSENTER_EIP MSR, 2-374 A32_SYSENTER_EIP MSR, 2-375 A32_SYSENTER_EIP MSR, 2-375 A32_THERM_LISTERUPT MSR, 2-377 A32_THERM_STATUS MSR, 2-377 A32_SYSENTER_EIP MSR, 2-377 A32_THERM_STATUS MSR, 2-377 A32_SYSENTER_EIP MSR, 2-377 A32_MSC, ROSE MSR, 2-79, 2-90, 2-95, 2-146, 2-183 A32_SYSENTER_EIP MSR, 2-375 A32_THERM_STATUS MSR, 2-375 A32_THERM_STATUS MSR, 2-377 A32_MSC, ROSE MSR, 2-79, 2-90, 2-95, 2-146, 2-183 A32_MSC, ROSE MSR, 2-376 A32_SYSENTER_EIP MSR, 2-377 A32_MSC, ROSE MSR, 2-376 A32_THERM_STATUS MSR, 2-375 A32_THERM_STATUS MSR, 2-375 A32_MSC, ROSE MSR, 2-376 A32_SYSENTER_EIP MSR, 2-377 A32_MSC, ROSE MSR, 2-376 A32_THERM_STATUS MSR, 2-377 A32_MSC, ROSE MSR, 2-376 A32_SYSENTER_EIP MSR, 2-377 A32_MSC, ROSE MSR, 2-376 A32_SYSENTER_EIP MSR, 2-377 A32_MSC, ROSE MSR, 2-376 A32_SYSENTER_EIP MSR, 2-377 A32_MSC, ROSE MSR, 2-377 A32_MSC, ROSE MSR, 2-376 A32_SYSENTER_EIP MSR, 2-377 A32_MSC, ROSE MSR, 2-376 A32_SYSENTER_EIP MSR, 2-377 A32_MSC, ROSE MSR, 2-376 A32_SYSENTER_EIP MSR, 2-377 A32_MSC, ROSE MSR, 2-376 A32_SYSENTER_EIP		= '
Capability MSRs See VMX capability MSRs See VMX capability MSRs Compatibility Software, 1-5 Context, task (see Task state) Debug store (see CPL) Debug store (see DS) Debug store (s	Byte order, 1-5	
Capability MSRs See VMX capability MSRs See Task state) CPUID instruction Syntax for data, 1-7 Current privilege level (see CPL) Debug store (see DS) See DS (debug store) mechanism See DS (debug store) mech		IA32_SYSENTER_EIP MSR, 2-374
Capability MSRs See VMX capability MSRs See Task state) CPUID instruction Syntax for data, 1-7 Current privilege level (see CPL) Debug store (see DS) See DS (debug store) mechanism See DS (debug store) mech		IA32_SYSENTER_ESP MSR, 2-373
Capability MSRs See VMX cRX cRX capability MSRs See VMX cRX capability MSRs	C	
See VMX capability MSRs IA3Z_THERM_STATUS MSR, 2-377 Cornext task (see Task state) IA3Z_TME_STAMP, COUNTER MSR, 2-370 Context task (see Task state) IA3Z_TME_STAMP, COUNTER MSR, 2-370 Context task (see Task state) IA3Z_MSR, 2-79, 2-90, 2-99, 2-155, 2-193, 2-391, 2-404 Syntax for data, 1-7 IA3Z_MSR, 2-79, 2-90, 2-99, 2-155, 2-193, 2-391, 2-405 DEBUGTIMSR MSR, 2-419 IA3Z_MSR, 2-79, 2-90, 2-99, 2-155, 2-193, 2-391, 2-405 DEBUGTIMSR MSR, 2-419 IA3Z_MSR, 2-79, 2-90, 2-99, 2-155, 2-193, 2-391, 2-405 Saz_MSR, 2-79, 2-90, 2-99, 2-155, 2-193, 2-391, 2-405 Saz_MSR, 2-79, 2-90, 2-99, 2-155, 2-193, 2-391, 2-404 Saz_MSR, 2-79, 2-90, 2-99, 2-155, 2-193, 2-391, 2-404	Canability MSRs	
Compatibility software, 1-5 IA3Z_TIME_STAMP_COUNTER MSR, 2-370 2-90, 2-90, 2-155, 2-193, 2-391, 2-404 IA3Z_MX_CRS, 2-770 g. 2-90, 2-99, 2-155, 2-193, 2-391, 2-404 IA3Z_MX_CRS, 2-770 g. 2-90, 2-99, 2-155, 2-193, 2-391, 2-405 IA3Z_MX_CRS, 2-770 g. 2-90, 2-99, 2-155, 2-193, 2-391, 2-405 IA3Z_MX_CRS, 2-770 g. 2-90, 2-99, 2-155, 2-193, 2-391, 2-405 IA3Z_MX_CRS, 2-770 g. 2-90, 2-99, 2-155, 2-193, 2-391, 2-405 IA3Z_MX_CRS, 2-770 g. 2-90, 2-99, 2-155, 2-193, 2-391, 2-405 IA3Z_MX_CRS, 1-70 g. 2-90, 2-99, 2-155, 2-193, 2-391, 2-405 IA3Z_MX_CRS, 1-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-70 g. 2-90, 2-99, 2-155, 2-192, 2-390, 2-404 IA3Z_MX_CRS, 1-70 g. 2-70 g. 2-90, 2-99, 2-155, 2-193, 2-290, 2-99, 2-155, 2-193, 2-290, 2-99, 2-155, 2-193, 2-290, 2-99, 2-155, 2-193, 2-290, 2-99, 2-155, 2-193, 2-290, 2-99, 2-155, 2-193, 2-290, 2-99, 2-155, 2-1		,
software, 1-5 Context task (see Task state) CPUID instruction syntax for data, 1-7 Current privilege level (see CPL) Debug store (see DS) Debug store (see		
Context, task (see Task state) (CPUID instruction syntax for data, 1-7 (Urrent privilege level (see CPL) D Debug store (see DS) DEBUGCTLMSR MSR, 2-419 Debugging facilities see DS (debug store) mechanism E Exceptions notation, 1-7 G G Gilobal descriptor table register (see GDTR) Global descriptor table (see GDT) H Hexadecimal numbers, 1-6 H Hexadecimal numbers, 1-6 IA32_MIX_GRA_SIKED, MSR, 2-79, 2-90, 2-99, 2-155, 2-192, 2-390, 2-90, 2-90, 2-91, 2-99, 2-155, 2-192, 2-390, 2-90, 2-90, 2-90, 2-91, 2-90, 2-90, 2-90, 2-91, 2-90, 2-90, 2-91, 2-90, 2-90, 2-91, 2-90, 2-90, 2-91, 2-90, 2-90, 2-90, 2-91, 2-90, 2		IA32_TIME_STAMP_COUNTER MSR, 2-370
CPUID instruction syntax for data, 1-7 Current privilege level (see CPL) Debug store (see DS) Debug store (see DT) Debug store (see DT		IA32_VMX_BASIC MSR, 2-79, 2-90, 2-98, 2-154, 2-192, 2-390, 2-404
CPUID instruction syntax for data, 1-7 Current privilege level (see CPL) Debug store (see DS) Debug store (see DT) Debug store (see DT	Context, task (see Task state)	IA32 VMX CR0 FIXED0 MSR, 2-79, 2-90, 2-99, 2-155, 2-193, 2-391.
syntax for data, 1-7 Current privilege level (see CPL) Debug store (see DS) Debug store (see DS) Debug store (see DS) DEBUGCTLMSR MSR, 2-419 Debugging facilities see DS (debug store) mechanism E Exceptions notation, 1-7 G G Global descriptor table register (see GDTR) Global descriptor table register (see GDTR) Global descriptor table (see GDT) H H Hexadecimal numbers, 1-6 I I IA32_BIOS_SIGN_ID MSR, 2-370 IA32_BIOS_SIGN_ID MSR, 2-373 IA32_MSR, 2-374 IA32_DEBUGCTL MSR, 2-375 IA32_MC_GRAP MSR, 2-391 IA32_MC_GRAP MSR, 2-391 IA32_MC_GRAP MSR, 2-376 IA32_MC_RSR MSR, 2-376 IA32_MC_RSR MSR, 2-376 IA32_MC_RSR MSR, 2-375 IA32_MC_RSR MSR, 2-374 IA32_MC_RSR MSR, 2-375 IA32_MC_RSR MSR, 2-374 IA32_MC_RSR MSR, 2-375 IA32_MC_RSR MSR, 2-375 IA32_MC_RSR MSR, 2-376 IA32_MC_RSR MSR, 2-375		
Debug store (see DS) Debug store (see DS) DEBUGCTLMSR MSR, 2-419 Debugging facilities see DS (debug store) mechanism E Exceptions notation, 1-7 G G Global descriptor table register (see GDTR) Global descriptor table (see GDT) H H Hexadecimal numbers, 1-6 H HA Hexadecimal numbers, 1-6 I I I I I I I I I I I I I I I I I I		
Debug store (see DS) DEBUGCTLMSR MSR, 2-419 Debugging facilities see DS (debug store) mechanism E Exceptions notation, 1-7 G G Global descriptor table register (see GDTR) Global descriptor table (see GDT) H H Hexadecimal numbers, 1-6 I I IA32_AMC, BASE MSR, 2-370 IA32_BIOS_SIGN_ID MSR, 2-373 IA32_CLOCK_MODULATION MSR, 2-73, 2-85, 2-95, 2-146, 2-183, 2-358, IA32_MC, R1S, R2-37 IA32_DEBUGCTL MSR, 2-374 IA32_DEBUGCTL MSR, 2-375 IA32_MCG_CRAY MSR, 2-376 IA32_MCG_RAY MSR, 2-376 IA32_MCG_RAY MSR, 2-377 IA32_MCG_RAY MSR, 2-377 IA32_MCG_RAY MSR, 2-377 IA32_MCG_RAY MSR, 2-377 IA32_MCG_RAY MSR, 2-376 IA32_MCG_RAY MSR, 2-375 IA32_MCG_RAY MSR, 2-376 IA32_MCG_RAY MSR, 2-376 IA32_MCG_RAY MSR, 2-375 IA32_MCG_RAY MSR, 2-376 IA32_MCG_RAY MSR, 2-375 IA32_MCG_RAY MSR, 2-376 IA32_MCG_RAY MSR, 2-376 IA32_MCG_RAY MSR, 2-375 IA32_MCG_RAY MSR, 2-376 IA32_MCG_RAY MSR, 2-376 IA32_MCG_RAY MSR, 2-376 IA32_MCG_RAY MSR, 2-376 IA32_MCG_RAY MSR, 2-375 IA32_MCG_RAY MSR, 2-376 IA32		
Debug store (see DS) DEBUGCTLMSR MSR, 2-419 Debugging facilities see DS (debug store) mechanism E Exceptions notation, 1-7 G G Global descriptor table register (see GDTR) Global descriptor table (see GDT) H H Hexadecimal numbers, 1-6 I 1 I 1A32_MCB_RSE MSR, 2-370 IA32_BOS_UPDT_TRIC MSR, 2-373 IA32_BOS_UPDT_TRIC MSR, 2-373 IA32_BOS_UPDT_TRIC MSR, 2-373 IA32_BOS_UPDT_TRIC MSR, 2-374 IA32_DDS_AREA MSR, 2-381 IA32_DS_AREA MSR, 2-381 IA32_DS_AREA MSR, 2-381 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-374 IA32_MCG_RSM MSR, 2-374 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-377 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-374 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-374 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-374 IA32_MCG_RSM MSR, 2-374 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-375 IA	Current privilege level (see CPL)	
Debug store (see DS) DEBUGCTLMSR MSR, 2-419 Debugging facilities see DS (debug store) mechanism E Exceptions notation, 1-7 G G Global descriptor table register (see GDTR) Global descriptor table (see GDT) H H Hexadecimal numbers, 1-6 I 1 I 1A32_MCB_RSE MSR, 2-370 IA32_BOS_UPDT_TRIC MSR, 2-373 IA32_BOS_UPDT_TRIC MSR, 2-373 IA32_BOS_UPDT_TRIC MSR, 2-373 IA32_BOS_UPDT_TRIC MSR, 2-374 IA32_DDS_AREA MSR, 2-381 IA32_DS_AREA MSR, 2-381 IA32_DS_AREA MSR, 2-381 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-374 IA32_MCG_RSM MSR, 2-374 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-377 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-374 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-374 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-374 IA32_MCG_RSM MSR, 2-374 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-375 IA32_MCG_RSM MSR, 2-376 IA32_MCG_RSM MSR, 2-375 IA		IA32_VMX_CR4_FIXED0 MSR, 2-79, 2-90, 2-99, 2-155, 2-193, 2-391,
Debug store (see DS) DEBUGCTLMSR MSR, 2-419 Debugging facilities see DS (debug store) mechanism E Exceptions notation, 1-7 G G Global descriptor table register (see GDTR) Global descriptor table (see GDT) H Hexadecimal numbers, 1-6 H Hexadecimal numbers, 1-6 I IA32_MIC_BASE MSR, 2-370 IA32_BIOS_UPDT_TRIC MSR, 2-373 IA32_BIOS_UPDT_TRIC MSR, 2-374 IA32_BIOS_UPDT_TRIC MSR, 2-374 IA32_DEBUGCTL MSR, 2-381 IA32_DEBUGCTL MSR, 2-381 IA32_DEBUGCTL MSR, 2-381 IA32_DEBUGCTL MSR, 2-374 IA32_DEBUGCTL MSR, 2-375 IA32_MIC_RSER MSR, 2-376 IA32_MIC_RSER MSR, 2-376 IA32_MIC_MSC MSR, 2-375 IA32_MIC_MSC MSR, 2-375 IA32_MIC_MSC MSR, 2-375 IA32_MIC_MSC MSR, 2-376 IA32_MIC_MSC MSR, 2-376 IA32_MIC_MSC MSR, 2-375 IA32_MIC_RSER MSR, 2-376 IA32_MIC_RSER MSR, 2-376 IA32_MIC_RSER MSR, 2-376 IA32_MIC_RSER MSR, 2-374 IA32_MIC_RSER MSR, 2-375 IA32_MIC_RSER MSR, 2-375 IA32_MIC_RSER MSR, 2-376 IA32_MIC_RSER MSR, 2-376 IA32_MIC_RSER MSR, 2-375 IA32_MIC_RSER MSR, 2-376 IA32_MIC_RSER MSR, 2-376 IA32_MIC_RSER MSR, 2-375 IA32_MIC_RSER MSR, 2-374 IA32_MIC_RSER MSR, 2-375 IA32_MIC_RSER MSR, 2-376 IA32_MIC_RSER MSR, 2-376 IA32_MIC_RSER MSR, 2-375 IA32_MIC_RSER MSR, 2-376 IA32_MIC_RSER MSR, 2-375 IA32_MIC_RSER MSR, 2-376 IA32_MIC_RSER MSR, 2-375 IA32_MIC_RSER MSR, 2-375 IA32_MIC_RSER MSR, 2-376 IA32_MIC_RSER MSR, 2-375 IA32_MIC_RSER	D.	
Debug store (see US) Debugding facilities See DS (debug store) mechanism E Exceptions notation, 1-7 G G G Global descriptor table register (see GDTR) Global descriptor table (see GDT) H Hexadecimal numbers, 1-6 Hexadecimal numbers, 1-6 I I Ja32_MCB_CRS MSR, 2-370 JA32_BOS_UPD_T TRIC MSR, 2-373 JA32_BOS_UPD_T TRIC MSR, 2-373 JA32_BOS_UPD_T TRIC MSR, 2-373 JA32_BOS_UPD_T TRIC MSR, 2-374 JA32_DS_AREA MSR, 2-381 JA32_DS_AREA MSR, 2-381 JA32_DS_AREA MSR, 2-381 JA32_MCG_RSR MSR, 2-375 JA32_MCG_RSR MSR, 2-376 JA32_MCG_RSR MSR, 2-374 JA32_MCG_RSR MSR, 2-375 JA32_MCG_RSR MSR, 2-376 JA32_MCG_RSR MSR, 2-376 JA32_MCG_RSR MSR, 2-376 JA32_MCG_RSR MSR, 2-374 JA32_MCG_RSR MSR, 2-375 JA32_MCG_RSR MSR, 2-376 JA32_MCG_RSR MSR, 2-376 JA32_MCG_RSR MSR, 2-374 JA32_MCG_RSR MSR, 2-375 JA32_MCG_RSR MSR, 2-376 JA32_MCG_RSR MSR, 2-376 JA32_MCG_RSR MSR, 2-375 JA32_MCG_RSR MSR, 2-376 JA32_MCG_RSR MSR, 2-376 JA32_MCG_RSR MSR, 2-375 JA32_MCG_RSR MSR, 2-374 JA32_MCG_RSR MSR, 2-375 JA32_MCG_RSR MSR, 2-375 JA32_MCG_RSR MSR, 2-376 JA32_MCG_RSR MSR, 2-375 JA32_MCG_RSR MSR, 2-375 JA32_MCG_RSR MSR, 2-375 JA32_MCG_RSR MSR, 2-376 JA32_MCG_RSR MSR, 2-375 JA32_MCG_RSR MSR, 2-375 JA32_MCG_RSR MSR, 2-374 JA32_MCG_RSR MSR, 2-375 JA32_MCG_RSR MSR, 2-375 JA32_MCG_RSR MSR, 2-375 JA32_MCG_RSR MSR, 2-376 JA32_MCG_RSR MSR, 2-375 JA32_MCG_RSR MSR, 2-374 JA32_MCG_RSR MSR, 2-375 JA32_MCG_RSR MSR, 2-375	U	
DEBUĞUTLMSR MSR, 2-419 Debugging facilities see DS (debug store) mechanism E Exceptions notation, 1-7 G GIobal descriptor table register (see GDTR) Global descriptor table see GDT) H H Hexadecimal numbers, 1-6 Hexadecimal numbers, 1-6 I Ja2_ PIC_ BASE MSR, 2-370 IA32_ PIC_ BASE MSR, 2-373 IA32_ BIOS_ UPDT_TRIG MSR, 2-733 IA32_ BIOS_ LOUCK, MODULATION MSR, 2-733 IA32_ BIOS_ LOUCK, MODULATION MSR, 2-734 IA32_ DEBUCTL MSR, 2-374 IA32_ DEBUCTL MSR, 2-374 IA32_ MCG_ CRA MSR, 2-375 IA32_ MCG_ RIS MSR, 2-376 IA32_ MCG_ RIS MSR, 2-374 IA32_ MCG_ RIS MSR, 2-374 IA32_ MCG_ RIS MSR, 2-375 IA32_ MCG_ RIS MSR, 2-376 IA32_ MCG_ RIS MSR, 2-374 IA32_ MCG_ RIS MSR, 2-375 IA32_ MCG_ RIS MSR, 2-376 IA32_ MCG_ RIS MSR, 2-376 IA32_ MCG_ RIS MSR, 2-376 IA32_ MCG_ RESKRVEDN, 2-375, 2-476 IA32_ MCG_ RESKRVEDN, 2-375,	Debug store (see DS)	
Debugging facilities see DS (debug store) mechanism E		,
A32_VMX_EXT_CTLS MSR, 2-79, 2-90, 2-99, 2-155, 2-192, 2-390, 2-404		
Exceptions notation, 1-7 G G Global descriptor table register (see GDTR) Global descriptor table (see GDT) HA32_VMZ_PROCEASED_CTLS MSR, 2-79, 2-90, 2-98, 2-154, 2-192, 2-390, 2-404 PA32_VMZ_PROCEASED_CTLS MSR, 2-79, 2-90, 2-91, 2-99, 2-100, 2-100, 2-154, 2-155, 2-192, 2-193, 2-228, 2-390, 2-404, 2-405 PA32_VMZ_PROCEASED_CTLS MSR, 2-79, 2-80, 2-90, 2-91, 2-99, 2-100, 2-154, 2-155, 2-192, 2-193, 2-228, 2-390, 2-404, 2-405 PA32_VMZ_PROCEASED_CTLS MSR, 2-79, 2-80, 2-90, 2-91, 2-99, 2-100, 2-154, 2-155, 2-192, 2-193, 2-228, 2-390, 2-404, 2-405 PA32_VMZ_PROCEASED_CTLS MSR, 2-390, 2-404, 2-405 PA32_VMZ_PROCEASED_CTLS MSR, 2-390, 2-90, 2-91, 2-99, 2-100, 2-154, 2-155, 2-192, 2-193, 2-201, 2-99, 2-100, 2-154, 2-155, 2-192, 2-193, 2-201, 2-99, 2-100, 2-154, 2-155, 2-192, 2-193, 2-201, 2-99, 2-100, 2-154, 2-155, 2-192, 2-193, 2-201, 2-99, 2-100, 2-154, 2-155, 2-192, 2-193, 2-291, 2-99, 2-100, 2-100, 2-154, 2-154, 2-192, 2-390, 2-404 PA32_VMZ_PROCEASED_CTLS MSR, 2-391 Instruction operations, 1-6 Intel (See GDT) PA32_MCL_ROLE PASS_PROCEASED_CTLS MSR, 2-390, 2-90, 2-98, 2-154, 2-192, 2-390, 2-404 PA32_VMZ_PROCEASED_CTLS MSR, 2-391, 2-390, 2-404 PA32_VMZ_PROCEASED_CTLS MSR, 2-391, 2-390, 2-404, 2-405 PA32_VMZ_PROCEASED_CTLS MSR, 2-391, 2-390, 2-404 PA32_VMZ_PROCEASED_CTLS MSR, 2-391, 2-390, 2-404, 2-405 PA32_VMZ_PROCEASED_CTLS MSR, 2-391, 2-390, 2-404 PA32_VMZ_PROCEASED_CTLS MSR, 2-370, 2-390, 2-404, 2-405 PA32_VMZ_PROCEASED_CTLS MSR, 2-390, 2-90, 2-90, 2-91, 2-99, 2-155, 2-192, 2-193, 2-291, 2-390, 2-404, 2-405 PA32_VMZ_PROCEASED_CTLS MSR, 2-390, 2-404 PA32_VMZ_PROCEASED_CTLS MSR, 2-390, 2-404, 2-405 PA32_VMZ_PROCEASED_CTLS MSR, 2-390, 2-4		2-404
E Exceptions notation, 1-7 E Exceptions notation, 1-7 G G G Global descriptor table register (see GDTR) Global descriptor table (see GDT) H H Hexadecimal numbers, 1-6 I I I I S J S J S J S J S J S J S J S J	see DS (debug store) mechanism	IA32 VMX EXIT CTLS MSR. 2-79. 2-90. 2-99. 2-155. 2-192. 2-390.
Exceptions notation, 1-7 Gas and descriptor table register (see GDTR) Global descriptor table register (see GDTR) Global descriptor table (see GDT) H H Hexadecimal numbers, 1-6 I I I I I I I I I I I I I I I I I I I		
Exceptions notation, 1-7 I Global descriptor table register (see GDTR) Global descriptor table (see GDT) Hexadecimal numbers, 1-6 Hexadecimal numbers, 1-6 I I I I I I I I I I I I I I I I I I I		
2.390, 2-404 (Base_motation, 1-7) (Base_motation, 1-8) (Base_motation of, 1-4 (Base_motation of, 1-4	E	M32_VMV_DINDASCD_CTLS_MSD_0.70_0.00_0.454_0.400
notation, 1-7 G	Exceptions	
G Global descriptor table register (see GDTR) Global descriptor table register (see GDTR) Global descriptor table (see GDT) Instruction operands, 1-6 Intel 64 architecture definition of, 1-4 relation to In-32, 1-4 Intel core Solo and Duo processors model-specific registers, 2-396 Intel NetBurst microarchitecture, 1-3 Intel software network link, 1-8 Intel VTune Performance Analyzer related information, 1-8 Intel VTune		2-390, 2-404
## Canal Company of the Company of t	notation, 1-7	IA32_VMX_PROCBASED_CTLS MSR, 2-79, 2-80, 2-90, 2-91, 2-99, 2-100
Global descriptor table register (see GDTR) Global descriptor table (see GDT) H Hexadecimal numbers, 1-6 I A32_APIC_BASE MSR, 2-370 IA32_BIOS_SIGN_ID MSR, 2-373 IA32_BIOS_UPDT_TRIC MSR, 2-373 IA32_CTL MSR, 2-374 IA32_DBUGTL MSR, 2-381 IA32_DBUGTL MSR, 2-394 IA32_MCG_REX MSR, 2-376 IA32_MCG_REX MSR, 2-376 IA32_MCG_REX MSR, 2-376 IA32_MCG_REX MSR, 2-377 IA32_MCG_REX MSR, 2-374 IA32_MCG_REX MSR, 2-375 IA32_MCG_REX MSR, 2-374 IA32_MCG_REX MSR, 2-374 IA32_MCG_REX MSR, 2-374 IA32_MCG_REX MSR, 2-375 IA32_MCG_REX MSR, 2-374 IA32_MCG_REX MSR, 2-374 IA32_MCG_REX MSR, 2-375 IA32_MCG_REX MSR, 2-374 IA32_MCG_REX MSR, 2-375 IA32_MCG_REX MSR, 2-374 IA32_MCG_REX MSR, 2-375 IA32_MCG_REX MSR, 2-375 IA32_MCG_REX MSR, 2-375 IA32_MCG_REX MSR, 2-374 IA32_MCG_REX MSR, 2-375 IA32_		2-154, 2-155, 2-192, 2-193, 2-228, 2-390, 2-404, 2-405
Global descriptor table register (see GDTR) Global descriptor table (see GDT) H Hexadecimal numbers, 1-6 Hexadecimal numbers, 1-6 Instruction operands, 1-6 Intel 64 architecture definition of, 1-4 relation to IA-32, 1-4 Intel Core Solo and Duo processors model-specific registers, 2-396 Intel NetBurst microarchitecture, 1-3 Intel software network link, 1-8 Intel software network link, 1-8 Intel VTune Performance Analyzer related information, 1-8 Intel Xeon processor, 1-1 Interrupts Interrupts Intel Core Solo and Duo processors model-specific registers, 2-396 Intel NetBurst microarchitecture, 1-3 Intel Software network link, 1-8 Intel VTune Performance Analyzer related information, 1-8 Intel Xeon processor, 1-1 Interrupts Interrupt descriptor table register (see IDTR) interrupt descriptor table register (see IDTR) Interrupt descriptor table (see IDT) IPI (see interprocessor interrupt) IPI (see interprocessor interrupt) Interrupt descriptor table (see IDT) IPI (see interprocessor interrupt) Interrupt descriptor table register (see IDTR) Interrupts Interrupt descriptor table (see IDTR) Interrupts Interrupt descriptor table register (see IDTR) Interrupts Interrupt descriptor table register (see IDTR) Interrupts Interrupt descriptor table (see IDTR) Interrupts Interrupt descriptor table (see IDTR) Interrupts Interrupt descriptor table register (see IDTR) Interrupts In	G	
Intel 64 architecture definition of, 1-4 relation to IA-32, 1-4		
definition of, 1-4 relation to IA-32, 1-4 Intel Core Solo and Duo processors model-specific registers, 2-396 Intel NetBurst microarchitecture, 1-3 Intel software network link, 1-8 Intel VTune Performance Analyzer related information, 1-8 Intel VTune Performance Analyzer related information, 1-8 Intel VTune Performance Analyzer related information, 1-8 Intel Van processor, 1-1 Interrupt Interrupt Interrupt descriptor table register (see IDTR) interrupt descriptor table (see IDT) IPI (see interprocessor interrupt) L Last branch record stack, 2-381, 2-382, 2-391 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R11 MSR, 2-376 IA32_MCG_R15 MSR, 2-377 IA32_MCG_R15 MSR, 2-377 IA32_MCG_RRX MSR, 2-374 IA32_MCG_RRX MSR, 2-374 IA32_MCG_RRX MSR, 2-374 IA32_MCG_RRX MSR, 2-375 IA32_MCG_RRX MSR, 2-374 IA32_MCG_RESERVEDD, 2-375, 2-476 IA32_MCG_RESERVEDD, 2-375, 2-476 M M M M M M M M M M M M M	Global descriptor table register (see GDTR)	
Hexadecimal numbers, 1-6 Hexadecimal numbers, 1-6 I I IA32_APIC_BASE MSR, 2-370 IA32_BIOS_SIGN_ID MSR, 2-373 IA32_BIOS_UPDT_TRIG MSR, 2-373 IA32_BIOS_UPDT_TRIG MSR, 2-77, 2-400, 2-409 IA32_OEBUGCTL MSR, 2-374 IA32_DEBUGCTL MSR, 2-381 IA32_DESUGC_CAP MSR, 2-374 IA32_MCG_CAP MSR, 2-376 IA32_MCG_R11 MSR, 2-376 IA32_MCG_R1 MSR, 2-376 IA32_MCG_R15 MSR, 2-377 IA32_MCG_RSEXPED, 2-375, 2-476 IA32_MCG_RSEXPED, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 M relation to IÅ-32, 1-4 Intel Core Solo and Duo processors model-specific registers, 2-396 Intel NetBurst microarchitecture, 1-3 Intel Software network link, 1-8 Intel VTune Performance Analyzer related information, 1-8 Intel VXeno processor, 1-1 Interrupts interrupt descriptor table register (see IDTR) interrupt descriptor table register (see IDTR) Interrupt descriptor table register (see IDTR) Interrupt descriptor table (see IDT) IPI (see interprocessor interrupt) L Last branch record stack, 2-381, 2-382, 2-391 Local descriptor table (see LDTR)	Global descriptor table (see GDT)	
Hexadecimal numbers, 1-6 Intel Core Solo and Duo processors model-specific registers, 2-396 Intel NetBurst microarchitecture, 1-3 Intel software network link, 1-8 Intel VTune Performance Analyzer related information, 1-8 Intel Xeon processor, 1-1 Interrupts interrupt descriptor table register (see IDTR) interrupt descriptor table register (see IDTR) interrupt descriptor table (see IDT) IPI (see interprocessor interrupt) Last branch record stack, 2-381, 2-382, MCG_R10 MSR, 2-376 IA32_MCG_R11 MSR, 2-376 IA32_MCG_R15 MSR, 2-374 IA32_MCG_R8X MSR, 2-374 IA32_MCG_R8X MSR, 2-374 IA32_MCG_R8X MSR, 2-374 IA32_MCG_R8X MSR, 2-375 IA32_MCG_R8X MSR, 2-374 IA32_MCG_R8X MSR, 2-375 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-375 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-375 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-375 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-375 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-375 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-375 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-375 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-376 I	, ,	definition of, 1-4
Hexadecimal numbers, 1-6 Intel Core Solo and Duo processors model-specific registers, 2-396 Intel NetBurst microarchitecture, 1-3 Intel software network link, 1-8 Intel VTune Performance Analyzer related information, 1-8 Intel Xeon processor, 1-1 Interrupts interrupt descriptor table register (see IDTR) interrupt descriptor table register (see IDTR) interrupt descriptor table (see IDT) IPI (see interprocessor interrupt) Last branch record stack, 2-381, 2-382, MCG_R10 MSR, 2-376 IA32_MCG_R11 MSR, 2-376 IA32_MCG_R15 MSR, 2-374 IA32_MCG_R8X MSR, 2-374 IA32_MCG_R8X MSR, 2-374 IA32_MCG_R8X MSR, 2-374 IA32_MCG_R8X MSR, 2-375 IA32_MCG_R8X MSR, 2-374 IA32_MCG_R8X MSR, 2-375 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-375 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-375 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-375 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-375 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-375 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-375 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-375 IA32_MCG_R8X MSR, 2-376 IA32_MCG_R8X MSR, 2-376 I		relation to IA-32. 1-4
Hexadecimal numbers, 1-6 I I IA32_APIC_BASE MSR, 2-370 IA32_BIOS_SIGN_ID MSR, 2-373 IA32_BIOS_UPDT_TRIG MSR, 2-373 IA32_BIOS_UPDT_TRIG MSR, 2-373 IA32_BIOS_UPDT_TRIG MSR, 2-72, 2-85, 2-95, 2-146, 2-183, 2-358, 2-377, 2-400, 2-409 IA32_CTL MSR, 2-374 IA32_DEBUGCTL MSR, 2-381 IA32_DEBUGCTL MSR, 2-391 IA32_MCG_CAP MSR, 2-375 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R10 MSR, 2-374 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R10 MS	H	
Intel NetBurst microarchitecture, 1-3 Intel software network link, 1-8 Intel software network link, 1-8 Intel VTune Performance Analyzer related information, 1-8 Intel Xeon processor, 1-1 Interrupts interrupt descriptor table register (see IDTR) interrupt descriptor table (see IDTR) interrupt descriptor table (see IDTR) interrupt descriptor table (see IDTR) IPI (see interprocessor interrupt) L L Last branch record stack, 2-381, 2-382, 2-391 LA32_MCG_MISC MSR, 2-376 LA32_MCG_R10 MSR, 2-376 LA32_MCG_R11 MSR, 2-376 LA32_MCG_R11 MSR, 2-376 LA32_MCG_R15 MSR, 2-377 LOcal descriptor table register (see LDTR) Local descriptor table register (see LDTR) Local descriptor table (see LDT) LVT (see Local vector table) M L LVT (see Local vector table)	Hovadocimal numbers 1.6	
Intel software network link, 1-8 Intel VTune Performance Analyzer related information, 1-8 Intel VTune Performance Analyzer related information, 1-8 Intel Xeon processor, 1-1 Interrupts Interrupt descriptor table register (see IDTR) interrupt descriptor table (see IDT) IPI (see interprocessor interrupt) Lasz_MCG_MISC MSR, 2-374 IA32_MCG_R10 MSR, 2-375 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R11 MSR, 2-376 IA32_MCG_R10 MSR, 2-374 IA32_MCG_R10 MSR, 2-374 IA32_MCG_R10 MSR, 2-374 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R10 MSR, 2-374 IA32_MCG_R10 MSR, 2-375 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R10 MSR, 2-375 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R10 MSR, 2-375 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R10 MSR, 2-375 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R10 MSR, 2-375 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R10 MSR, 2-375 IA32_MCG_R10 MSR, 2-375 IA32_MCG_R10 MSR, 2-375 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R10 MSR, 2-375 IA32_MCG_R1	Hexadecima numbers, 1-0	
Intel VTune Performance Analyzer related information, 1-8 Intel Xeon processor, 1-1 Interrupts interrupt descriptor table register (see IDTR) Interrupt descriptor table (see IDT) IPI (see interprocessor interrupt) IA32_MCG_CAP MSR, 2-374 IA32_MCG_CAP MSR, 2-375 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R11 MSR, 2-376 IA32_MCG_R8X MSR, 2-374 IA32_MCG_R8X MSR, 2-374 IA32_MCG_RBX MSR, 2-375 IA32_MCG_RBX MSR, 2-374 IA32_MCG_RBX MSR, 2-374 IA32_MCG_RESERVEDn, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476		
IA32_APIC_BASE MSR, 2-370 IA32_BIOS_SIGN_ID MSR, 2-373 IA32_BIOS_UPDT_TRIG MSR, 2-373 IA32_BIOS_UPDT_TRIG MSR, 2-373 IA32_CLOCK_MODULATION MSR, 2-72, 2-85, 2-95, 2-146, 2-183, 2-358, 2-377, 2-400, 2-409 IA32_CTL MSR, 2-374 IA32_DEBUGCTL MSR, 2-381 IA32_DEBUGCTL MSR, 2-381 IA32_MCG_CAP MSR, 2-374 IA32_MCG_RISC MSR, 2-375 IA32_MCG_RI1 MSR, 2-376 IA32_MCG_RI1 MSR, 2-376 IA32_MCG_RI1 MSR, 2-376 IA32_MCG_RAX MSR, 2-377 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RESERVEDn, 2-375, 2-476 IA32_MCG_RESERVEDn, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 M IA32_MCG_RFLAGS MSR, 2-375, 2-476 IMADEMORPHICAL MSR	T	
IA32_APIC_BASE MSR, 2-370 IA32_BIOS_SIGN_ID MSR, 2-373 IA32_BIOS_UPDT_TRIG MSR, 2-373 IA32_BIOS_UPDT_TRIG MSR, 2-373 IA32_CLOCK_MODULATION MSR, 2-72, 2-85, 2-95, 2-146, 2-183, 2-358, 2-377, 2-400, 2-409 IA32_CTL MSR, 2-374 IA32_DEBUGCTL MSR, 2-381 IA32_DEBUGCTL MSR, 2-381 IA32_MCG_CAP MSR, 2-374 IA32_MCG_RISC MSR, 2-375 IA32_MCG_RI1 MSR, 2-376 IA32_MCG_RI1 MSR, 2-376 IA32_MCG_RI1 MSR, 2-376 IA32_MCG_RAX MSR, 2-377 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RESERVEDn, 2-375, 2-476 IA32_MCG_RESERVEDn, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 M IA32_MCG_RFLAGS MSR, 2-375, 2-476 IMADEMORPHICAL MSR	•	Intel VTune Performance Analyzer
IA32_BIOS_SIGN_ID MSR, 2-373 IA32_BIOS_UPDT_TRIG MSR, 2-373 IA32_BIOS_UPDT_TRIG MSR, 2-373 IA32_CLOCK_MODULATION MSR, 2-72, 2-85, 2-95, 2-146, 2-183, 2-358, 2-377, 2-400, 2-409 IA32_CTL MSR, 2-374 IA32_DEBUGCTL MSR, 2-381 IA32_DEBUGCTL MSR, 2-381 IA32_DESUGCTL MSR, 2-391 IA32_MCG_CAP MSR, 2-374 IA32_MCG_RISC MSR, 2-375 IA32_MCG_RIO MSR, 2-376 IA32_MCG_RI1 MSR, 2-376 IA32_MCG_RIS MSR, 2-377 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RESERVEDn, 2-375, 2-476 IA32_MCG_RESERVEDn, 2-375, 2-476 IA32_MCG_RESERVEDN, 2-375, 2-476 IA32_MCG_RESERVEDN, 2-375, 2-476 IMMINISTRICT INTERPOLATION	IA32_APIC_BASE MSR, 2-370	
IA32_BIOS_UPDT_TRIG MSR, 2-373 IA32_CLOCK_MODULATION MSR, 2-72, 2-85, 2-95, 2-146, 2-183, 2-358, 2-377, 2-400, 2-409 IA32_CTL MSR, 2-374 IA32_DEBUGCTL MSR, 2-381 IA32_DS_AREA MSR, 2-391 IA32_MCG_CAP MSR, 2-374 IA32_MCG_R10 MSR, 2-375 IA32_MCG_R11 MSR, 2-376 IA32_MCG_R15 MSR, 2-377 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RESERVEDn, 2-375, 2-476	IA32 BIOS SIGN ID MSR. 2-373	•
IA32_CLOCK_MODULATION MSR, 2-72, 2-85, 2-95, 2-146, 2-183, 2-358, 2-377, 2-400, 2-409 IA32_CTL MSR, 2-374 IA32_DEBUGCTL MSR, 2-381 IA32_DS_AREA MSR, 2-391 IA32_MCG_CAP MSR, 2-374 IA32_MCG_R10 MSR, 2-375 IA32_MCG_R11 MSR, 2-376 IA32_MCG_R11 MSR, 2-376 IA32_MCG_R3X MSR, 2-374 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RBX MSR, 2-375 IA32_MCG_RBX MSR, 2-374 IA32_MCG_RBX MSR, 2-375, 2-476 IA32_MCG_RESERVEDN, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476		•
2-377, 2-400, 2-409 IA32_CTL MSR, 2-374 IA32_DEBUGCTL MSR, 2-381 IA32_DS_AREA MSR, 2-391 IA32_MCG_CAP MSR, 2-374 IA32_MCG_MISC MSR, 2-375 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R11 MSR, 2-376 IA32_MCG_R15 MSR, 2-377 IA32_MCG_R3X MSR, 2-374 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RSERVEDn, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 IMM INDEX TO TABLE (see IDT) IA32_MCG_RSERVEDN, 2-375, 2-476 IMM INDEX TO TABLE (see IDT) IA32_MCG_RSERVEDN, 2-375, 2-476 IMM INDEX TO MSR, 2-374 IMM INDEX TO MSR, 2-374 IMM INDEX TO MSR, 2-374 IMM INDEX TO MSR, 2-375, 2-476 IMM INDEX TO MSR, 2-374 IMM INDEX TO MSR, 2-374 IMM INDEX TO MSR, 2-374 IMM INDEX TO MSR, 2-375 INDEX TO MSR, 2-375 INDEX TO MSR, 2-375 INDEX TO MSR, 2-375 INDEX TO MSR, 2-376	,	
IA32_CTL MSR, 2-374 IA32_DEBUGCTL MSR, 2-381 IA32_DS_AREA MSR, 2-391 IA32_MCG_CAP MSR, 2-374 IA32_MCG_MISC MSR, 2-375 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R11 MSR, 2-376 IA32_MCG_R15 MSR, 2-377 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RESERVEDn, 2-375, 2-476 IA32_MCG_RESERVEDn, 2-375, 2-476 IA32_MCG_RESERVEDN IA32_MCG_RESE		, , , , , , , , , , , , , , , , , , , ,
IA32_DEBUGCTL MSR, 2-381 IA32_DS_AREA MSR, 2-391 IA32_MCG_CAP MSR, 2-374 IA32_MCG_MISC MSR, 2-375 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R11 MSR, 2-376 IA32_MCG_R15 MSR, 2-377 IA32_MCG_R15 MSR, 2-377 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RBX MSR, 2-374 IA32_MCG_RBX MSR, 2-375, 2-476 IA32_MCG_RESERVEDn, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 M M M M M M M M M M M M M		interrupt descriptor table (see IDT)
IA32_DEBUGCTL MSR, 2-381 IA32_DS_AREA MSR, 2-391 IA32_MCG_CAP MSR, 2-374 IA32_MCG_MISC MSR, 2-375 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R11 MSR, 2-376 IA32_MCG_R15 MSR, 2-377 IA32_MCG_R15 MSR, 2-377 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RBX MSR, 2-374 IA32_MCG_RESERVEDn, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 IA32_MCG_RESERVEDN,	IA32_CTL MSR, 2-374	IPI (see interprocessor interrupt)
IA32_MCG_CAP MSR, 2-374 IA32_MCG_MISC MSR, 2-375 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R11 MSR, 2-376 IA32_MCG_R15 MSR, 2-377 IA32_MCG_RAS MSR, 2-377 IA32_MCG_RAS MSR, 2-374 IA32_MCG_RBS MSR, 2-374 IA32_MCG_RESERVEDn, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 IA32_MCG_RELAGS MSR, 2-375, 2-476 M M M M M M M M M M M M M	IA32_DEBUGCTL MSR, 2-381	` ' '
IA32_MCG_CAP MSR, 2-374 IA32_MCG_MISC MSR, 2-375 IA32_MCG_R10 MSR, 2-376 IA32_MCG_R11 MSR, 2-376 IA32_MCG_R15 MSR, 2-377 IA32_MCG_RAS MSR, 2-377 IA32_MCG_RAS MSR, 2-374 IA32_MCG_RBS MSR, 2-374 IA32_MCG_RESERVEDn, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 IA32_MCG_RELAGS MSR, 2-375, 2-476 M M M M M M M M M M M M M	IA32 DS AREA MSR. 2-391	
Has2_MCG_MISC MSR, 2-375 Last branch record stack, 2-381, 2-382, 2-391 Local descriptor table register (see LDTR) Local descriptor table (see LDT) LVT (see Local vector table) LVT (see Lo	,	L
R32_MCG_R10 MSR, 2-376 record stack, 2-381, 2-382, 2-391 R32_MCG_R11 MSR, 2-376 Local descriptor table register (see LDTR) R32_MCG_R15 MSR, 2-377 Local descriptor table (see LDT) R32_MCG_RAX MSR, 2-374 LVT (see Local vector table) R32_MCG_RESERVEDN, 2-375, 2-476 M R32_MCG_RFLAGS MSR, 2-375, 2-476 M		
IA32_MCG_R11 MSR, 2-376 IA32_MCG_R15 MSR, 2-377 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RESERVEDn, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 IA32_MCG_RESERVED MSR, 2-375, 2-476		
IA32_MCG_R15 MSR, 2-377 IA32_MCG_RAX MSR, 2-374 IA32_MCG_RBX MSR, 2-374 IA32_MCG_RESERVEDn, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 M M M M M M M M M M M M M		
IA32_MCG_RAX MSR, 2-374 IA32_MCG_RBX MSR, 2-374 IA32_MCG_RESERVEDn, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 M M M M M M M M M M M M M	IA32_MCG_R11 MSR, 2-376	Local descriptor table register (see LDTR)
IA32_MCG_RAX MSR, 2-374 IA32_MCG_RBX MSR, 2-374 IA32_MCG_RESERVEDn, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 M M M M M M M M M M M M M	IA32_MCG_R15 MSR, 2-377	Local descriptor table (see LDT)
IA32_MCG_RBX MSR, 2-374 IA32_MCG_RESERVEDn, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 M M M M M M M M M M M M M		, ,
IA32_MCG_RESERVEDn, 2-375, 2-476 IA32_MCG_RFLAGS MSR, 2-375, 2-476 M M M M M M M M M M M M M		201 (See Edeal Vector Lable)
IA32_MCG_RFLAGS MSR, 2-375, 2-476		
Manageritus 1958, 2-373, 2-470		M
IA32_MCG_RIP MSR, 2-375, 2-476 Memory type range registers (see MTRRs)	IA32_MCG_RFLAGS MSR, 2-375, 2-476	
	IA32_MCG_RIP MSR, 2-375, 2-476	Memory type range registers (see MTRRs)

INDEX

Model-specific registers (see MSRs) Modes of operation (see Operating modes) MSRs architectural, 2-3	Segmented addressing, 1-6 System-management mode (see SMM)
list of, 2-1 P6 family processors, 2-412 Pentium 4 processor, 2-67, 2-81, 2-209, 2-225, 2-243, 2-369, 2-395 Pentium processors, 2-421, 2-501	Translation lookaside buffer (see TLB)
MSR_DEBUGCTLA MSR, 2-381 MSR_DEBUGCTLB MSR, 2-75, 2-87, 2-96, 2-149, 2-186, 2-227, 2-361, 2-402, 2-411 MSR_EBC_FREQUENCY_ID MSR, 2-372, 2-373	U Uncached (UC) memory type (see Strong uncached (UC) memory type)
MSR_EBC_HARD_POWERON MSR, 2-370 MSR_EBC_SOFT_POWERON MSR, 2-371, 2-443 MSR_LASTBRANCH_TOS, 2-381 MSR_LASTBRANCH_0_TO_IP, 2-393 MSR_LASTBRANCH_n MSR, 2-382, 2-446	V VMX capability MSRs IA32_VMX_BASIC MSR, 2-79, 2-90, 2-98, 2-154, 2-192, 2-390,
MSR_LASTBRANCH_n_FROM_IP MSR, 2-391 MSR_LASTBRANCH_n_TO_LIP MSR, 2-393 MSR_LER_FROM_LIP MSR, 2-381 MSR_LER_TO_LIP MSR, 2-381	2-404 IA32_VMX_CRO_FIXED0 MSR, 2-79, 2-90, 2-99, 2-155, 2-193, 2-391, 2-404 IA32_VMX_CRO_FIXED1 MSR, 2-79, 2-90, 2-99, 2-155, 2-193,
MSR_PEBS_MATRIX_VERT MSR, 2-388, 2-478 MSR_PLATFORM_BRV, 2-380, 2-481	2-391, 2-405 IA32_VMX_CR4_FIXED0 MSR, 2-79, 2-90, 2-99, 2-155, 2-193, 2-391, 2-405 IA32_VMX_CR4_FIXED1 MSR, 2-79, 2-90, 2-91, 2-99, 2-155,
N NetBurst microarchitecture (see Intel NetBurst microarchitecture) Nonmaskable interrupt (see NMI) Notation	2-193, 2-391, 2-405 IA32_VMX_ENTRY_CTLS MSR, 2-79, 2-90, 2-99, 2-155, 2-192, 2-391, 2-404 IA32_VMX_EXIT_CTLS MSR, 2-79, 2-90, 2-99, 2-155, 2-192,
bit and byte order, 1-5 conventions, 1-5 exceptions, 1-7 hexadecimal and binary numbers, 1-6	2-390, 2-404 IA32_VMX_MISC MSR, 2-79, 2-90, 2-99, 2-155, 2-193, 2-391, 2-404 IA32_VMX_PINBASED_CTLS MSR, 2-79, 2-90, 2-98, 2-154,
Instructions operands, 1-6 reserved bits, 1-5 segmented addressing, 1-6	2-192, 2-390, 2-404 IA32_VMX_PROCBASED_CTLS MSR, 2-79, 2-80, 2-90, 2-91, 2-99, 2-100, 2-154, 2-155, 2-192, 2-193, 2-228, 2-390, 2-404, 2-405
	IA32_VMX_VMCS_ENUM MSR, 2-391
Operands	W
Operands instruction, 1-6	WC buffer (see Write combining (WC) buffer)
P5_MC_ADDR MSR, 2-67, 2-81, 2-92, 2-141, 2-178, 2-353, 2-396, 2-405, 2-412, 2-422 P5_MC_TYPE MSR, 2-67, 2-81, 2-92, 2-141, 2-178, 2-353, 2-396,	
2-406, 2-412, 2-422 P6 family processors description of, 1-1 MSR supported by, 2-412	
Page frame (see Page) PDBR (see CR3 control register) PEBS_UNAVAILABLE flag IA32_MISC_ENABLE MSR, 2-379	
Pentium 4 processor, 1-1 MSRs supported, 2-67, 2-81, 2-92, 2-109, 2-111, 2-135, 2-139, 2-369, 2-395 Pentium II processor, 1-3	
Pentium III processor, 1-3 Pentium M processor MSRs supported by, 2-405	
Pentium Pro processor, 1-3 Pentium processor, 1-1 MSR supported by, 2-421 Precise event-based sampling (see PEBS)	
R	
Related literature, 1-8 Requested privilege level (see RPL) Reserved bits, 1-5	