BIOS and Kernel Developer's Guide (BKDG) for AMD Family 14h Models 00h-0Fh Processors

Table of Contents

List of Figures

List of Tables

Revision History

Revision 3.00: Initial public release.

1 Overview

The AMD Family 14h Models 00h-0Fh processor (in this document referred to as *the processor*) is a processing unit that supports x86-based instruction sets. The processor includes (a) up to two independent central processing unit cores (referred to as *cores*), (b) one PCIe® root complex with generation 2 link support, (c) one 64-bit double-data rate 3 (DDR3) system memory DRAM interface, and optionally (d) a graphics core.

AMD Family 14h processors are distinguished by the combined ExtendedFamily and BaseFamily fields of the CPUID instruction (see CPUID Fn8000_0001_EAX).

1.1 Intended Audience

This document provides the processor behavioral definition and associated design notes. It is intended for platform designers and for programmers involved in the development of low-level BIOS (basic input/output system) functions, drivers, and operating system kernel modules. It assumes prior experience in personal computer platform design, microprocessor programming, and legacy x86 and AMD64 microprocessor architecture. The reader should also have familiarity with various platform technologies, such as DDR3 DRAM.

1.2 Reference Documents

- Advanced Configuration and Power Interface (ACPI) Specification (www.acpi.info)
- AMD Family 14h Processor Power and Thermal Data Sheet, #45410.
- AMD Voltage Regulator Specification, #40182
- AMD Socket FT1 Functional Data Sheet, #44444
- AMD64 Architecture Programmer's Manual Volume 1: Application Programming, #24592
- AMD64 Architecture Programmer's Manual Volume 2: System Programming, #24593
- AMD64 Architecture Programmer's Manual Volume 3: Instruction-Set Reference, #24594
- AMD64 Architecture Programmer's Manual Volume 4: 128-Bit and 256-Bit Media Instructions, #26568
- AMD64 Architecture Programmer's Manual Volume 5: 64-Bit Media and x87 Floating-Point Instructions, #26569
- CPUID Specification, #25481
- Electrical Data Sheet for AMD Family 14h Models 00h-0Fh Processors, #44446
- PCI local bus specification (www.pcisig.org)
- PCI Express® specification (www.pcisig.org)
- Revision Guide for AMD Family 14h Models 00h-0Fh Processors, #47534
- System Management Bus (SMBus) specification (www.smbus.org)

1.3 Conventions

1.3.1 Numbering

- **Binary numbers**. Binary numbers are indicated by appending a “b” at the end, e.g., 0110b.
- **Decimal numbers**. Unless specified otherwise, all numbers are decimal. This rule does not apply to the register mnemonics described in section 3.1 [Register Descriptions and Mnemonics]; register mnemonics all use hexadecimal numbering.
- **Hexadecimal numbers**. Hexadecimal numbers are indicated by appending an “h” to the end, e.g., 45F8h.
- **Underscores in numbers**. Underscores are used to break up numbers to make them more readable. They do not imply any operation. E.g., 0110_1100b.
- **Undefined digit**. An undefined digit, in any radix, is notated as “x”.

1.3.2 Arithmetic And Logical Operators

In this document, formulas follow some Verilog conventions for logic equations.

Table 1: Arithmetic and Logical Operators

Operator	Definition
{}	Curly brackets are used to indicate a group of bits that are concatenated together. Each set of bits is separated by a comma. For example: {Addr[3:2], Xlate[3:0]} represents a 6-bit value; the two MSBs are Addr[3:2] and the four LSBs are Xlate[3:0].
|	Bitwise OR operator. For example: 01b | 10b == 11b.
||	Logical OR operator. For example: 01b || 10b == 1b; logical OR treats a multibit operand as 1 if the operand is >=1 and produces a 1-bit result.
&	Bitwise AND operator. For example: 01b & 10b == 00b.
&&	Logical AND operator. For example: 01b && 10b == 1b; logical AND treats a multibit operand as 1 if the operand is >=1 and produces a 1-bit result.
^	Bitwise exclusive-OR operator; sometimes used as "raised to the power of" as well, as indicated by the context in which it is used. For example: 01b ^ 10b == 11b. 2^2 == 4.
~	Bitwise NOT operator (also known as one's complement). For example: ~10b == 01b.
!	Logical NOT operator. For example: !10b == 0b; logical NOT treats a multibit operand as 1 if the operand is >=1 and produces a 1-bit result.
==	Logical "is equal to" operator.
!=	Logical "is not equal to" operator.
<=	Less than or equal operator.
>=	Greater than or equal operator.
<<	Shift left first operand by the number of bits specified by the second operand. For example: 01b << 1 = 10b.
>>	Shift right first operand by the number of bits specified by the second operand. For example: 10b >> 1 = 01b.
*****	Arithmetic multiplication operator.
/	Arithmetic division operator.
[]	Square brackets are used to indicate a range of values and can take two forms. The unit of the range is implied by context. (E.g. register bit index, register number.) 1. [y:x]: A contiguous range of values is specified, from x being the least significant to y being the most significant. 2. [z, ..., y, x]: A comma separated list of values is specified, from x being the least significant to z being the most significant.

Table 2: Functions

Function	Definition
ABS	ABS(integer-expression): Remove sign from a signed value.
FLOOR	FLOOR(integer-expression): Rounds a real number down to the nearest integer.
CEIL	CEIL(real-expression): Rounds a real number up to the nearest integer.
MIN	MIN(integer-expression-list): Picks the minimum integer or real value of a comma separated list.

Table 2: Functions

Function	Definition
MAX	MAX(integer-expression-list): Picks the maximum integer or real value of a comma separated list.
COUNT	COUNT(integer-expression): Returns the number of binary 1's in the integer.
ROUND	ROUND(real-expression): Rounds to the nearest integer; halfway rounds away from zero.

The order in which logical operators are applied is: ~ first, & second, and | last.

For example, the equation:

```
Output[3:0] = {A[1:0], B[3:2]} & C[3:0] | ~D[3:0] & E[9:6],
```

is translated as:

```
Output[3] = (A[1] & C[3]) | (~D[3] & E[9]);
Output[2] = (A[0] & C[2]) | (~D[2] & E[8]);
Output[1] = (B[3] & C[1]) | (~D[1] & E[7]);
Output[0] = (B[2] & C[0]) | (~D[0] & E[6]);
```

1.4 Definitions

Table 3: Definitions

Term	Definition
AP	Application processor. See 2.3 [Processor Initialization].
BCS	Base configuration space. See 2.7 [Configuration Space].
BIST	Built-in self-test. Hardware within the processor that generates test patterns and verifies that they are stored correctly (in the case of memories) or received without error (in the case of links).
Boot VID	Boot voltage ID. This is the VDDCR_CPU and VDDCR_NB voltage level that the processor requests from the external voltage regulator during the initial phase of the cold boot sequence.
BSP	Boot strap processor. See 2.3 [Processor Initialization].
C-states	These are ACPI-defined CPU power states. C0 is operational. All other C-states are low-power states in which the processor is not executing code. See 2.5.3.2 [C-states].
CAF	C-state action field. See D18F4x118 and D18F4x11C.
Canonical address	An address in which the state of the most-significant implemented bit is duplicated in all the remaining higher-order bits, up to bit 63.
Channel	See DRAM channel.
CMP	Chip multi-processing. Refers to processors that include multiple cores. See 2.1 [Processor Overview].
Coherent fabric	The coherent fabric includes the DRAM controller and caches of the system. See 2.2 [System Overview].
COF	Current operating frequency of a given clock domain. See 2.5.3.1 [Core P-states].
Cold reset	PWROK is de-asserted and RESET_L is asserted. See 2.3 [Processor Initialization].
CPU or core	The instruction execution unit of the processor. See 2.1 [Processor Overview].
CpuCoreNum	Specifies the core number. See 2.4.2 [Processor Cores and Downcoring].
CPUID function X	Refers to the CPUID instruction when EAX is preloaded with X. See 3.19 [CPUID Instruction Registers].
CS	Chip select. See D18F2x[4C:40] [DRAM CS Base Address].
DCT	DRAM controller. See 2.9 [DRAM Controller (DCT)].
DDI	Digital display interface. See 2.13 [Digital Display Interface].
DEV	DMA exclusion vector. See 2.8.3 [DMA Exclusion Vectors (DEV)].
DID	Divisor identifier. Specifies the post-PLL divisor used to reduce the COF. See 2.5.3.1 [Core P-states].
Display refresh	Traffic used for display refresh in UMA systems.
DMA	Direct memory access. An access to memory that does not use the CPU.
Doubleword or DW	A 32-bit value.
Downcoring	Removal of cores. See 2.4.2 [Processor Cores and Downcoring].
DRAM channel	The part of the DRAM interface that connects to a 64-bit DIMM. See 2.9 [DRAM Controller (DCT)].
DS	Downstream. Refers to the direction of data on a link. In the context of a memory buffer link, this refers to data flow from the controller to the memory buffer.

Table 3: Definitions

Term	Definition
Dual-Plane	Refers to a processor or system board where VDDCR_CPU and VDDCR_NB are separate and may operate at independent voltage levels. See 2.5.1 [Processor Power Planes And Voltage Control].
ECS	Extended configuration space. See 2.7 [Configuration Space].
EDS	Electrical data sheet. See 1.2 [Reference Documents].
FCH	Fusion controller hub. The platform device that contains the bridge to the system BIOS.
FDS	Functional data sheet; there is one FDS for each package type.
FEQ	Front-end queue.
FID	Frequency identifier. Specifies the PLL frequency multiplier for a given clock domain. See 2.5.3.1 [Core P-states].
Ganged	A PCIe® link or voltage regulator in which all portions are controlled as one.
GB or Gbyte	Gbyte or Gigabyte; 1,073,741,824 bytes.
#GP	A general-protection exception.
#GP(0)	Notation indicating a general-protection exception (#GP) with error code of 0.
GPU	Graphics processing unit. A GPU may be internal (on-chip) or external (off-chip).
GpuEnabled	GpuEnabled = (D1F0x00 != FFFF_FFFFh).
GT/s	Giga-transfers per second.
HTC	Hardware thermal control. See 2.10.3.1 [PROCHOT_L and Hardware Thermal Control (HTC)].
HTC-active state	Hardware-controlled lower-power, lower-performance state used to reduce temperature. See 2.10.3.1 [PROCHOT_L and Hardware Thermal Control (HTC)].
IBS	Instruction based sampling. See 3 [Registers].
IFQ	In-flight queue.
IO configuration	Access to configuration space through IO ports CF8h and CFCh. See 2.7 [Configuration Space].
IORR	IO range register. See MSRC001_00[18,16] [IO Range Registers Base (IORR_BASE[1:0])].
KB	Kilobyte; 1024 bytes.
L1 caches	The level 1 caches of the core including the instruction cache and the data cache.
L2 cache	The level 2 cache of each core.
Linear (virtual) address	The address generated by a core after the segment is applied.
Link	Generic term that refers to a PCIe® link.
LINT	Local interrupt.
Logical address	The address generated by a core before the segment is applied.
LOW_LATENCY	Used in an IF statement to define when low latency operation is required (e.g. to support a high speed isochronous channel or flash memory swap file). Low latency operation typically requires additional power, so this is not the default configuration.
LVT	Local vector table. A collection of APIC registers that define interrupts for local events. E.g., APIC[530:500] [Extended Interrupt [3:0] Local Vector Table].
Master abort	This is a PCI-defined term that is applied to transactions on other than PCI buses. It indicates that the transaction is terminated without affecting the intended target; reads return all 1's; writes are discarded; the master abort error code is returned in the response, if applicable; master abort error bits are set if applicable.

Table 3: Definitions

Term	Definition
MB	Megabyte; 1024 KB.
MEMCLK	Refers to the clock signals, M[3:0]_CLK, that are driven from the processor to DDR DIMMs. The MEMCLK frequency is determined by D18F2x94[MemClkFreq].
Micro-op	Instructions have variable-length encoding and many perform multiple primitive operations. The processor does not execute these complex instructions directly. Instead, the processor decodes them internally into simpler fixed-length instructions called macro-ops. The processor scheduler breaks down macro-ops into sequences of even simpler instructions called micro-ops, each of which specifies a single primitive operation.
MMIO	Memory-mapped input-output range. This is physical address space that is mapped to the IO functions such as the links or MMIO configuration. The link MMIO ranges are specified by D18F1x[B8,B0,A8,A0,98,90,88,80] [Memory Mapped IO Base].
MMIO configuration	Access to configuration space through memory space. See 2.7 [Configuration Space].
MOF	Maximum operating frequency of the core(s). Normally this is the core COF in P-state 0. See 2.5.3.1 [Core P-states].
MRS	Mode register set. A command used to access the mode registers of DDR3 SDRAM.
MSI	Message signaled interrupts. Refer to the PCI Express® specification for more information.
MSR	Model-specific register. The CPU includes several MSRs for general configuration and control. See 3.20 [MSRs - MSR0000_xxxx] for the beginning of the MSR register definitions.
MTRR	Memory-type range register. The MTRRs specify the type of memory associated with various memory ranges. See MSR0000_00FE, MSR0000_020[E,C,A,8,6,4,2,0], MSR0000_02[6F:68,59,58,50], and MSR0000_02FF.
NB	Northbridge. The transaction routing block of the node. See 2.1 [Processor Overview].
NBCIF	NB to core interface.
NCLK	The main northbridge clock. The NCLK frequency is the NB COF.
Node	See 2.1 [Processor Overview].
Normalized address	Addresses used by the DCT. See 2.9 [DRAM Controller (DCT)].
Octword	A 128-bit value.
ODM	On-DIMM mirroring. See D18F2x[4C:40][OnDimmMirror].
ODT	On-die termination, which is applied to DRAM interface signals.
Operational frequency	The frequency at which the processor operates. See 2.5 [Power Management].
PCIe®	PCI Express®.
PDS	Product data sheet.
Physical address	Addresses used by cores in transactions sent to the NB.
PRBS	Pseudo-random bit sequence.
Processor	See 2.1 [Processor Overview].
P-state	Performance state. See 2.5 [Power Management].
PTE	Page table entry.
PVI	Parallel VID interface. See 2.5.1 [Processor Power Planes And Voltage Control].
Quadword	A 64-bit value.

Table 3: Definitions

Term	Definition
REG	Used in an IF statement to define a register that has a unique definition based on the register address. See D0F0x64_x5[B,9,7,5] for an example.
Root complex	See the PCI Express® 2.0 Base Specification.
RX	Receiver.
SBI	Sideband Interface. See 2.17 [Sideband Interface (SBI)].
SCLK	Internal GPU clock.
Shutdown	A state in which the affected core waits for either INIT, RESET, or NMI. When shutdown state is entered, a shutdown special cycle is sent on the IO links.
Slam	To change the voltage to a new value in one step (as opposed to stepping). See 2.5.1.5.1 [Hardware-Initiated Voltage Transitions].
SMAF	System management action field. This is the code passed from the SMC to the processors in STPCLK assertion messages. The action taken by the processors in response to this message is specified by D18F3x80 [ACPI Power State Control Low].
SMBus	System management bus. Refers to the protocol on which the serial VID interface (SVI) commands and SBI are based. See 2.5.1 [Processor Power Planes And Voltage Control], 2.17 [Sideband Interface (SBI)], and 1.2 [Reference Documents].
SMC	System management controller. This is the platform device that communicates system management state information to the processor through an IO link, typically the FCH.
SMI	System management interrupt. See 2.4.6.2.1 [SMM Overview].
SMM	System management mode. See 2.4.6.2 [System Management Mode (SMM)].
Speculative event	A performance monitor event counter that counts all occurrences of the event even if the event occurs during speculative code execution.
STC	Software thermal control.
SVI	Serial VID interface. See 2.5.1 [Processor Power Planes And Voltage Control].
SVM	Secure virtual machine. See 2.4.7 [Secure Virtual Machine Mode (SVM)].
TCC	Temperature calculation circuit. See 2.10 [Thermal Functions].
TDP	Thermal design power. A power consumption parameter that is used in conjunction with thermal specifications to design appropriate cooling solutions for the processor.
TSC	Time stamp counter.
TX	Transmitter.
UI	Unit interval. This is the amount of time equal to one half of a clock cycle.
UMI	Unified Media Interface. The link between the processor and the FCH.
US	Upstream. Refers to the direction of data on a link.
us	Microsecond.
VDDCR_CPU	Main power supply to the processor core logic.
VDDCR_NB	Main power supply to the processor NB logic.
VID	Voltage level identifier. See 2.5.1 [Processor Power Planes And Voltage Control].
Virtual CAS	The clock in which CAS is asserted for the burst, N, plus the burst length (in MEMCLKs), minus 1; so the last clock of virtual CAS = N + (BL/2) - 1.
VRM	Voltage regulator module.
Warm reset	RESET_L is asserted only (while PWROK stays high). See 2.3 [Processor Initialization].

Table 3: Definitions

Term	Definition
WDT	Watchdog timer. A timer that detects activity and triggers an error if a specified period of time expires without the activity. For example, see MSRC001_0074 [CPU Watchdog Timer (CpuWdTmrCfg)] or the NB watchdog timer in D18F3x40.
Word	A 16-bit value.
XBAR	Cross bar; command packet switch. See 2.8.1 [Northbridge (NB) Architecture].

1.5 Changes Between Revisions and Product Variations

1.5.1 Revision Conventions

The processor revision is specified by CPUID Fn0000_0001_EAX[Model]. This document uses a revision letter instead of specific model numbers. The following table shows the relationship between revision and model. All behavior marked with a revision letter apply to future revisions unless they are superseded by a change in a later revision. See the *Revision Guide for AMD Family 14h Models 00h-0Fh Processors* for additional information about revision determination.

Table 4: Processor revision conventions

Revision	CPUID Fn0000_0001_EAX[Model]
B	1h

2 Functional Description

2.1 Processor Overview

The *processor* is a package that contains (1) one to two cores, (2) one PCIe® root complex with generation 2 link support, (3) one 64-bit DDR3 interface for communication to system memory, and (4) one communication packet routing block referred to as the *northbridge* (NB).

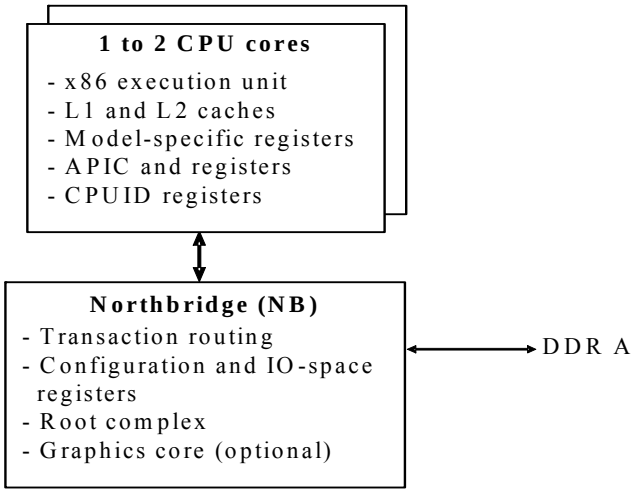


Figure 1: A processor

Each core includes x86 instruction execution logic, a first-level (L1) data cache, a first-level instruction cache, and a second level (L2) general-purpose cache. There is a set of model-specific registers (MSRs) and APIC registers associated with each core. Processors that include multiple cores are said to incorporate *chip multi-processing* or CMP.

The links are input-output links, as defined by the PCI Express® Base Specification.

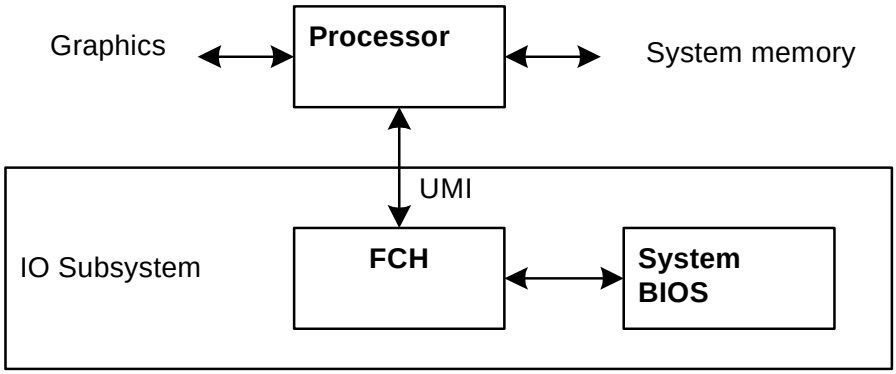
The DRAM interface supports a 64-bit DDR3 unbuffered DIMM channel.

The northbridge routes transactions between the cores, the links, the graphics core and the DRAM interface. It includes the configuration register space for the device.

2.2 System Overview

The following diagram illustrates the expected system architecture:

Figure 2: System Diagram



2.3 Processor Initialization

This section describes the initialization sequence after a cold reset.

Core 0 of the processor is the boot strap processor (BSP) and begins executing code at the reset vector. The remaining core, if supported, does not fetch code until its enable bit is set. See D18F0x68[Cpu1En].

2.3.1 BSP Initialization

The BSP must perform the following tasks as part of the boot process:

- Store BIST information from the EAX register into an unused processor register.
- If supported, determine the type of startup from either the keyboard controller or D18F0x6C[InitDet]. If this boot sequence was caused by an INIT, BIOS can branch away from the cold/warm reset initialization path.
- Determine the history of this reset using D18F0x6C[ColdRstDet].
  - If this is a cold reset, BIOS must clear the MCi_STATUS MSRs. See Table 36.
  - If this is a warm reset, BIOS may check for valid MCA errors and if present save the status for later use. See 2.16.2.4 [Handling Machine Check Exceptions].
- Enable the cache, program the MTRRs for Cache-as-RAM and initialize the Cache-as-RAM, as described in 2.3.3 [Cache Initialization For General Storage During Boot].
- Configure the local APIC. See 2.4.6.1.1 [ApicId Enumeration Requirements].
- Configure the I/O links. See 2.11.2.2 [Link Configurations].
- Configure the DRAM controller. See 2.9.3 [DCT/DRAM Initialization and Resume].
- Perform device enumeration for all I/O-link devices (see link specification).
- Configure the processor power management. See 2.5 [Power Management].
- If supported, allow the other processor core to begin fetching instructions by programming D18F0x68[Cpu1En]=1.

2.3.2 AP Initialization

Processor cores other than core 0 begin executing code from the reset vector. They must perform the following tasks as part of the boot process:

- Store BIST information from the EAX register into an unused processor register.
- If supported, determine the type of startup from either the keyboard controller or D18F0x6C[InitDet]. If this

boot sequence was caused by a an INIT, BIOS can branch away from the cold/warm reset initialization path.
- Determine the history of this reset using D18F0x6C [ColdRstDet].
  - If this is a cold reset, BIOS must clear the MCi_STATUS MSRs. See Table 36.
  - If this is a warm reset, BIOS may check for valid MCA errors and if present save the status for later use. See 2.16.2.4 [Handling Machine Check Exceptions].
- Configure the local APIC. See 2.4.6.1.1 [ApicId Enumeration Requirements].
- Configure the processor power management. See 2.5 [Power Management].

2.3.3 Cache Initialization For General Storage During Boot

Prior to initializing the DRAM controller for system memory, BIOS may use the L2 cache of each core as general storage.

The L2 cache as storage is described as follows:
- Each core has its own L2 cache.
- BIOS manages the mapping of the L2 storage such that cacheable accesses do not cause L2 victims.
- The L2 size, L2 associativity, and L2 line size is determined by reading CPUID Fn8000_0006_ECX[L2Size, L2Assoc, L2LineSize]. L2WayNum is defined to be the number of ways indicated by the L2Assoc code.
  - The L2 cache is viewed as (L2Size/L2LineSize) cache lines of storage, organized as L2WayNum ways, each way being (L2Size/L2WayNum) in size.
    - E.g. L2Assoc=8 so L2WayNum=16 (there are 16 ways). If L2Size=512KB then there are 16 blocks of cache, each 512KB/16 in size, or 32KB each.
  - For each of the following values of L2Size, the following values are defined:
    - L2Size=512KB: L2Tag=PhysAddr[35:15], L2WayIndex=PhysAddr[14:6].
  - PhysAddr[5:0] addresses the L2LineSize number of bytes of storage associated with the cache line.
  - The L2 cache, when allocating a line at L2WayIndex, will:
    - Pick an invalid way before picking a valid way.
    - Prioritize the picking of invalid ways such that way 0 is the highest priority and L2WayNum-1 is the lowest priority.
  - It is recommended that BIOS assume a simpler allocation of L2 cache memory, being L2WayNum size-aligned blocks of memory, each being L2Size/L2WayNum bytes.
  - BIOS can rely on a minimum L2Size of 512 KB for Family 14h Models 00h-0Fh. See CPUID Fn8000_0006_ECX[L2Size].

Memory types are supported as follows:
  - WP-IO: BIOS ROM may be assigned the write-protect IO memory type and may be accessed read-only as data and fetched as instructions.
    - WP-IO accesses, both read and write, do not get evicted to the L2 and therefore do not need to be considered for allocation into the L2.
  - WB-DRAM: General storage may be assigned the write-back DRAM memory type and may be accessed as read-write data, but not accessed by instruction fetch.
    - BIOS initializes an L2LineSize sized and aligned location in the L2 cache, mapped as write-back DRAM, with 1 read to at least 1 byte of the L2LineSize sized and aligned WB-DRAM address. BIOS may store to a line only after it has been allocated by a load.
    - Fills, sent to the disabled memory controller, return undefined data.
  - All of memory space that is not accessed as WP-IO or WB-DRAM space must be marked as UC memory type.

- In order to prevent victimizing L2 data, no more than L2WayNum cache lines accessed as WP-IO or WB-DRAM may have the same L2WayIndex.
  - Software does not need to know which ways the L2WayNum lines are allocated to for any given value of

L2WayIndex, only that invalid ways will be selected for allocation before valid ways will be selected for allocation.
- Software can deallocate a line in the L2 by using CLFLUSH, and thus allow for a cache line to be filled with a different location.

Performance monitor event PMCx07F [L2 Fill/Writeback], sub-event bit 1, titled “L2 Writebacks to system“, can be used to indicate whether L2 dirty data was victimized and sent to the disabled memory controller.

The following requirements must be satisfied prior to using the cache as general storage:
- Paging must be disabled.
- MSRC001_0015[INVD_WBINVD]=0.
- MSRC001_1020[DisStreamSt]=1.
- MSRC001_1021[DIS_SPEC_TLB_RLD]=1. Disable speculative ITLB reloads.
- MSRC001_1022[DIS_HW_PF]=1.
- INVD, and WBINVD must not be used during cache as general storage but may be used when tearing down cache-as-ram for all cores on a node.
- The BIOS must not use 3DNow!™, SSE, or MMX™ instructions, with the exception of the following list: MOVD, MOVQ, MOVDQA, MOVQ2DQ, MOVDQ2Q.
- BIOS must not enable exceptions, page-faults, and other interrupts.
- BIOS must not use software prefetches.

When BIOS is done using the cache as general storage the following steps are followed:
1. An INVD instruction should be executed on each core that used cache as general storage.
2. If DRAM is initialized and there is data in the cache that needs to get moved to main memory, CLFLUSH or WBINVD may be used instead of INVD, but software must ensure that needed data in main memory is not overwritten.
3. Program the following:
   - MSRC001_0015[INVD_WBINVD]=1.
   - MSRC001_1020[DisStreamSt]=0.
   - MSRC001_1021[DIS_SPEC_TLB_RLD]=0.
   - MSRC001_1022[DIS_HW_PF]=0.

2.3.4 BIOS Requirements For 64-Bit Operation

Refer to the AMD64 Architecture Programmer's Manual for a description of the 64-bit mode.

2.4 Processor Core

The majority of the behavioral definition of the processor core is specified in the AMD64 Architecture Programmer’s Manual. See 1.2 [Reference Documents].

2.4.1 Virtual Address Space

The processor supports 48 address bits of virtual memory space (256 terabyte) as indicated by CPUID Fn8000_0008_EAX.

2.4.2 Processor Cores and Downcoring

Each processor supports 1 to 2 cores.

- The number of cores supported by the processor is specified by D18F3xE8[CmpCap].
- The core number, *CpuCoreNum*, is provided to the software on each core through CPUID

Fn0000_0001_EBX[LocalApicId] and APIC20[ApicId]. CpuCoreNum also affects D18F0x68[Cpu1En].
- The boot core is always the core reporting CpuCoreNum = 0.

2.4.3 Access Type Determination

The access type determination and destination affects routing specified in section 2.8.4 [Northbridge Routing].

2.4.4 System Address Map

The processor core supports a 36 bit physical address that does not conflict with reserved link address regions.

2.4.4.1 Memory Access to the Physical Address Space

All memory accesses to the physical address space from a core are sent to its associated northbridge (NB). All memory accesses from a link are routed through the NB.

A core access to physical address space has two important attributes that the CPU must determine before issuing the access to the NB: the cache attribute (e.g., WB, WC, UC; as described in the MTRRs) and the access destination (DRAM or MMIO).

2.4.4.1.1 Determining The Cache Attribute

1. The CPU translates the logical address to a physical address. In that process it determines the initial cache attribute based on the settings of the Page Table Entry PAT bits, MSR0000_02FF [MTRR Default Memory Type (MTRRdefType)], MSR0000_020[E,C,A,8,6,4,2,0] [Variable-Size MTRRs (MTRRphysBasen)], and MSR0000_02[6F:68,59,58,50] [Fixed-Size MTRRs].
2. The ASeg and TSeg SMM mechanisms are then checked in parallel to determine if the initial cache attribute should be overridden (see MSRC001_0112 [SMM TSeg Base Address (SMMAddr)] and MSRC001_0113 [SMM TSeg Mask (SMMMask)]). If the address falls within an enabled ASeg/TSeg region, then the final cache attribute is determined as specified in MSRC001_0113.

This mechanism is managed by BIOS and does not require any setup or changes by system software.

2.4.4.1.2 Determining The Access Destination for CPU Accesses

The access destination, DRAM or MMIO, is based on the highest priority of the following ranges that the access falls in:

1. (Lowest priority) Compare against the top-of memory (TOM) registers (see MSRC001_001A, and MSRC001_001D).
2. The Fixed-Size MTRRs: MSR0000_02[6F:68,59,58,50].
3. The IORRs: MSRC001_00[18,16] and MSRC001_00[19,17].
4. (Highest priority) TSEG & ASEG: MSRC001_0112 and MSRC001_0113.

To determine the access destination, the following steps are taken:

1. The CPU compares the address against MSRC001_001A [Top Of Memory (TOP_MEM)], and MSRC001_001D [Top Of Memory 2 (TOM2)], to determine if the default access destination is DRAM or MMIO space.
2. For addresses below 1M byte, the address is then compared against the appropriate Fixed MTRRs to override the default access destination. Each fixed MTRR includes two bits, RdDram and WrDram, that determine the destination based on the access type. See MSR0000_02[6F:68,59,58,50].

3. The CPU then compares the address against the IORRs (MSRC001_00[18,16] and MSRC001_00[19,17]); if it matches, the default access destination is overridden as specified by the IORRs. BIOS can use the IORRs to create an IO hole within a range of addresses that would normally be mapped to DRAM. It can also use the IORRs to re-assert a DRAM destination for a range of addresses that fall within a bigger IO hole that overlays DRAM.
    a) Operating system software never needs to program IORRs to re-map addresses that naturally target DRAM; any such programming is done by BIOS.
4. The ASeg and TSeg SMM mechanisms are then checked in parallel to determine if the destination should be overridden (see MSRC001_0112 and MSRC001_0113). If the address falls within an enabled ASeg/TSeg region, then the destination is determined as specified in MSRC001_0113.

This mechanism is managed by BIOS and does not require any setup or changes by system software.

2.4.5 Timers

Each core includes the following timers. These timers do not vary in frequency regardless of the current P-state or C-state.

- MSR0000_0010 [Time Stamp Counter (TSC)]; the TSC increments at the rate specified by MSRC001_0015[TscFreqSel].
- The APIC timer (APIC380 and APIC390), which decrements at a rate of 2x CLKIN.

2.4.6 Interrupts

2.4.6.1 Local APIC

The local APIC contains logic to receive interrupts from a variety of sources and to send interrupts to other local APICs, as well as registers to control its behavior and report status. Interrupts can be received from:

- IO devices including the FCH (IO APICs)
- Other local APICs (inter-processor interrupts)
- APIC timer
- Thermal events
- Performance counters
- Legacy local interrupts from the FCH (INTR and NMI)
- APIC internal errors

The APIC timer, thermal events, performance counters, local interrupts, and internal errors are all considered local interrupt sources, and their routing is controlled by local vector table entries. These entries assign a message type and vector to each interrupt, allow them to be masked, and track the status of the interrupt.

IO and inter-processor interrupts have their message type and vector assigned at the source and are unaltered by the local APIC. They carry a destination field and a mode bit that together determine which local APIC(s) accepts them. The destination mode (DM) bit specifies if the interrupt request packet should be handled in physical or logical destination mode. If the destination field matches the broadcast value specified by D18F0x68[ApicExtBrdCst], then the interrupt is a broadcast interrupt and is accepted by all local APICs regardless of destination mode.

2.4.6.1.1 ApicId Enumeration Requirements

System hardware and BIOS must ensure that the number of cores per processor (NC) exposed to the operating system by all tables, registers, and instructions across all cores and processors in the system is identical.

See2.4.8.1 [Multi-Core Support] to derive NC.

Operating systems are expected to use CPUID Fn8000_0008_ECX[ApicIdCoreIdSize[3:0]], the number of least significant bits in the Initial APIC ID that indicate core ID within a processor, in constructing per-core CPUID masks. (ApicIdCoreIdSize[3:0] determines the maximum number of cores (MNC) that the processor could theoretically support, not the actual number of cores that are actually implemented or enabled on the processor, as indicated by CPUID Fn8000_0008_ECX[NC].) BIOS must use the ApicId MNC rule when assigning APIC20 [APIC ID][ApicId] values as described below.

ApicId MNC rule: The ApicId of core j must be enumerated/assigned as:

ApicId[core=j] = (OFFSET_IDX) * MNC + j

Where “OFFSET_IDX” is an integer offset (0 to N) used to shift up the CPU ApicId values to allow room for IOAPIC devices.

It is recommended that BIOS use the following APIC ID assignments for the broadest operating system support. Given N = (MNC) and M = (Number of IOAPICs):

- Assign the core APIC IDs first from 0 to N-1, and the IOAPIC IDs from N to N+(M-1).

2.4.6.1.2 Physical Destination Mode

The interrupt is only accepted by the local APIC whose APIC20[ApicId] matches the destination field of the interrupt. Physical mode allows up to 255 APICs to be addressed individually.

2.4.6.1.3 Logical Destination Mode

A local APIC accepts interrupts selected by APICD0 [Logical Destination] and the destination field of the interrupt using either cluster or flat format as configured by APICE0[Format].

If flat destinations are in use, bits 7-0 of APICD0[Destination] are checked against bits 7-0 of the arriving interrupt’s destination field. If any bit position is set in both fields, the local APIC is a valid destination. Flat format allows up to 8 APICs to be addressed individually.

If cluster destinations are in use, bits 7-4 of APICD0[Destination] are checked against bits 7-4 of the arriving interrupt’s destination field to identify the cluster. If all of bits 7-4 match, then bits 3-0 of APICD0[Destination] and the interrupt destination are checked for any bit positions that are set in both fields to identify processors within the cluster. If both conditions are met, the local APIC is a valid destination. Cluster format allows 15 clusters of 4 APICs each to be addressed.

2.4.6.1.4 Interrupt Delivery

SMI, NMI, INIT, Startup, and External interrupts are classified as non-vectored interrupts.

When an APIC accepts a non-vectored interrupt, it is handled directly by the processor instead of being queued in the APIC. When an APIC accepts a fixed or lowest-priority interrupt, it sets the bit in APIC[270:200] [Interrupt Request] corresponding to the vector in the interrupt. For local interrupt sources, this comes from the vector field in that interrupt’s local vector table entry. If a subsequent interrupt with the same vector arrives when the corresponding bit in APIC[270:200][RequestBits] is already set, the two interrupts are collapsed into one. Vectors 15-0 are reserved.

2.4.6.1.5 Vectored Interrupt Handling

APIC80 [Task Priority] and APICA0 [Processor Priority] each contain an 8-bit priority divided into a main priority (bits 7-4) and a priority sub-class (bits 3-0). The task priority is assigned by software to set a threshold priority at which the processor is interrupted.

The processor priority is calculated by comparing the main priority (bits 7-4) of APIC80[Priority] to bits 7-4 of the 8-bit encoded value of the highest bit set in APIC[170:100] [Interrupt In-Service]. The processor priority is the higher of the two main priorities.

The processor priority is used to determine if any accepted interrupts (indicated by APIC[270:200][RequestBits]) are high enough priority to be serviced by the processor. When the processor is ready to service an interrupt, the highest bit in APIC[270:200][RequestBits] is cleared, and the corresponding bit is set in APIC[170:100][InServiceBits]. The corresponding bit in APIC[1F0:180] [Trigger Mode] is set if the interrupt is level-triggered and cleared if edge-triggered.

When the processor has completed service for an interrupt, it performs a write to APICB0 [End of Interrupt], clearing the highest bit in APIC[170:100][InServiceBits] and causing the next-highest interrupt to be serviced. If the corresponding bit in APIC[1F0:180][TriggerModeBits] is set, a write to APICB0 is performed on all APICs to complete service of the interrupt at the source.

2.4.6.1.6 Interrupt Masking

Interrupt masking is controlled by the APIC410 [Extended APIC Control]. If APIC410[IerCap] is set, APIC[4F0:480] [Interrupt Enable] are used to mask interrupts. Any bit in APIC[4F0:480][InterruptEnableBits] that is clear indicates the corresponding interrupt is masked. A masked interrupt is not serviced and the corresponding bit in APIC[270:200][RequestBits] remains set.

2.4.6.1.7 Spurious Interrupts

In the event that the task priority is set to or above the level of the interrupt to be serviced, the local APIC delivers a spurious interrupt vector to the processor, as specified by APICF0 [Spurious Interrupt Vector]. APIC[170:100] is not changed and no write to APICB0 occurs.

2.4.6.1.8 Spurious Interrupts Caused by Timer Tick Interrupt

A typical interrupt is asserted until it is serviced. An interrupt is de-asserted when software clears the interrupt status bit within the interrupt service routine. Timer tick interrupt is an exception, since it is de-asserted regardless of whether it is serviced or not.

The processor is not always able to service interrupts immediately (i.e. when interrupts are masked by clearing EFLAGS.IM).

If the processor is not able to service the timer tick interrupt for an extended period of time, the INTR caused by the first timer tick interrupt asserted during that time is delivered to the local APIC in ExtInt mode and latched, and the subsequent timer tick interrupts are lost. The following cases are possible when the processor is ready to service interrupts:

- An ExtInt interrupt is pending, and INTR is asserted. This results in timer tick interrupt servicing. This occurs 50 percent of the time.
- An ExtInt interrupt is pending, and INTR is de-asserted. The processor sends the interrupt acknowledge cycle, but when the PIC receives it, INTR is de-asserted, and the PIC sends a spurious interrupt vector. This

occurs 50 percent of the time.

There is a 50 percent probability of spurious interrupts to the processor.

2.4.6.1.9 Lowest-Priority Interrupt Arbitration

Fixed, remote read, and non-vectored interrupts are accepted by their destination APICs without arbitration.

Delivery of lowest-priority interrupts requires all APICs to arbitrate to determine which one accepts the interrupt. If APICF0[FocusDisable] is clear, then the focus processor for an interrupt always accepts the interrupt. A processor is the focus of an interrupt if it is already servicing that interrupt (corresponding bit in APIC[170:100][InServiceBits] is set) or if it already has a pending request for that interrupt (corresponding bit in APIC[270:200][RequestBits] is set). If APIC410[IerCap] is set the interrupt must also be enabled in APIC[4F0:480][InterruptEnableBits] for a processor to be the focus processor. If there is no focus processor for an interrupt, or focus processor checking is disabled, then each APIC calculates an arbitration priority value, stored in APIC90 [Arbitration Priority], and the one with the lowest result accepts the interrupt.

The arbitration priority value is calculated by comparing APIC80[Priority] with the 8-bit encoded value of the highest bit set in APIC[270:200][RequestBits] (IRRVec) and the 8-bit encoded value of the highest bit set APIC[170:100][InServiceBits] (ISRVec). If APIC410[IerCap] is set the IRRVec and ISRVec are based off the highest enabled interrupt. The main priority bits 7-4 are compared as follows:

IF (APIC80[Priority[7:4]] >= IRRVec[7:4] and APIC80[Priority[7:4]] > ISRVec[7:4])
THEN APIC90[Priority] = APIC80[Priority]
ELSEIF (IRRVec[7:4] > ISRVec[7:4])
THEN APIC90[Priority] = {IRRVec[7:4],0h}
ELSE APIC90[Priority] = {ISRVect[7:4],0h}
ENDIF.

2.4.6.1.10 Inter-Processor Interrupts

APIC300 [Interrupt Command Low] and APIC310 [Interrupt Command High] provide a mechanism for generating interrupts in order to redirect an interrupt to another processor, originate an interrupt to another processor, or allow a processor to interrupt itself. A write to register APIC300 causes an interrupt to be generated with the properties specified by the APIC300 and APIC310 fields.

2.4.6.1.11 APIC Timer Operation

The local APIC contains a 32-bit timer, controlled by APIC320 [Timer Local Vector Table Entry], APIC380 [Timer Initial Count], and APIC3E0 [Timer Divide Configuration]. The processor bus clock is divided by the value in APIC3E0[Div] to obtain a time base for the timer. When APIC380[Count] is written, the value is copied into APIC390 [Timer Current Count]. APIC390[Count] is decremented at the rate of the divided clock. When the count reaches 0, a timer interrupt is generated with the vector specified in APIC320[Vector]. If APIC320[Mode] specifies periodic operation, APIC390[Count] is reloaded with the APIC380[Count] value, and it continues to decrement at the rate of the divided clock. If APIC320[Mask] is set, timer interrupts are not generated.

2.4.6.1.12 Generalized Local Vector Table

All LVTs (APIC320 through APIC370 and APIC[530:500]) support a generalized message type. The generalized values for MsgType are:
- 000b=Fixed

- 010b=SMI
- 100b=NMI
- 111b=ExtINT

2.4.6.1.13 State at Reset

At power-up or reset, the APIC is hardware disabled (MSR0000_001B[ApicEn]=0) so only SMI, NMI, INIT, and ExtInt interrupts may be accepted.

The APIC can be software disabled through APICF0[APICSWEn]. The software disable has no effect when the APIC is hardware disabled.

When a processor accepts an INIT interrupt, the APIC is reset as at power-up, with the exception that APIC20[ApicId], APIC410, and APIC[530:500] are unaffected.

2.4.6.2 System Management Mode (SMM)

System management mode (SMM) is typically used for system control activities such as power management. These activities are typically transparent to the operating system.

2.4.6.2.1 SMM Overview

SMM is entered by a core on the next instruction boundary after a system management interrupt (SMI) is received and recognized. A CPU may be programmed to broadcast a special cycle to the system, indicating that it is entering SMM mode. The core then saves its state into the SMM memory state save area and jumps to the SMI service routine (or SMI handler). The pointer to the SMI handler is specified by MSRs. The code and data for the SMI handler are stored in the SMM memory area, which may be isolated from the main memory accesses.

The core returns from SMM by executing the RSM instruction from the SMI handler. The core restores its state from the SMM state save area and resumes execution of the instruction following the point where it entered SMM. The core may be programmed to broadcast a special bus cycle to the system, indicating that it is exiting SMM mode.

2.4.6.2.2 Operating Mode and Default Register Values

The software environment after entering SMM has the following characteristics:

- Addressing and operation is in Real mode.
  - A far jump, call, or return in the SMI handler can only address the lower 1M of memory, unless the SMI handler first switches to protected mode.
  - If (MSRC001_0111[SmmBase]>=0010_0000h) then:
    - The value of the CS selector is undefined upon SMM entry.
    - The undefined CS selector value should not be used as the target of a far jump, call, or return.
- 4-Gbyte segment limits.
- Default 16-bit operand, address, and stack sizes (instruction prefixes can override these defaults).
- Control transfers that do not override the default operand size truncate the EIP to 16 bits.
- Far jumps or calls cannot transfer control to a segment with a base address requiring more than 20 bits, as in Real mode segment-base addressing, unless a change is made into protected mode.
- A20M# is disabled. A20M# assertion or de-assertion have no effect during SMM.
- Interrupt vectors use the Real mode interrupt vector table.
- The IF flag in EFLAGS is cleared (INTR is not recognized).

- The TF flag in EFLAGS is cleared.
- The NMI and INIT interrupts are masked.
- Debug register DR7 is cleared (debug traps are disabled).

The SMM base address is specified by MSRC001_0111 [SMM Base Address (SMM_BASE)][SmmBase]. Important offsets to the base address pointer are:
- MSRC001_0111[SmmBase] + 8000h: SMI handler entry point.
- MSRC001_0111[SmmBase] + FE00h - FFFFh: SMM state save area.

2.4.6.2.3 SMI Sources And Delivery

The processor accepts SMIs as link-defined interrupt messages only. The core destination of these SMIs is a function of the destination field of these messages. However, the expectation is that all such SMI messages are specified to be delivered globally (to all cores).

There are also several local events that can trigger SMIs. However, these local events do not generate SMIs directly. Each of them triggers a programmable IO cycle that is expected to target the SMI command port in the FCH and trigger a global SMI interrupt message back to the coherent fabric.

Local sources of SMI events that generate the IO cycle specified by MSRC001_0056 [SMI Trigger IO Cycle] are:

- In the core, as specified by:
  - MSRC001_0022 [Machine Check Exception Redirection].
  - MSRC001_00[53:50] [IO Trap (SMI_ON_IO_TRAP_[3:0])].
- All local APIC LVT registers programmed to generate SMIs.

The status for these is stored in SMMFEC4.

2.4.6.2.4 SMM Initial State

After storing the save state, execution starts at MSRC001_0111[SmmBase] + 08000h. The SMM initial state is specified in the following table.

Table 5: SMM initial state

Register	SMM Initial State
CS	SmmBase[19:4]
DS	0000h
ES	0000h
FS	0000h
GS	0000h
SS	0000h
General Purpose Registers	Unmodified
EFLAGS	0000_0002h
RIP	0000_0000_0000_8000h
CR0	Bits 0, 2, 3, and 31 cleared (PE, EM, TS, and PG); remainder is unmodified
CR4	0000_0000_0000_0000h
GDTR	Unmodified

Table 5: SMM initial state

Register	SMM Initial State
LDTR	Unmodified
IDTR	Unmodified
TR	Unmodified
DR6	Unmodified
DR7	0000_0000_0000_0400h
EFER	All bits are cleared except bit 12 (SVME) which is unmodified.

2.4.6.2.5 SMM Save State

In the following table, the offset field provides the offset from the SMM base address specified by MSRC001_0111 [SMM Base Address (SMM_BASE)].

Table 6: SMM Save State

Offset	Size	Contents		Access
FE00h	Word	ES	Selector	Read-only
FE02h	6 Bytes		Reserved	
FE08h	Quadword		Descriptor in memory format	
FE10h	Word	CS	Selector	Read-only
FE12h	6 Bytes		Reserved	
FE18h	Quadword		Descriptor in memory format	
FE20h	Word	SS	Selector	Read-only
FE22h	6 Bytes		Reserved	
FE28h	Quadword		Descriptor in memory format	
FE30h	Word	DS	Selector	Read-only
FE32h	6 Bytes		Reserved	
FE38h	Quadword		Descriptor in memory format	
FE40h	Word	FS	Selector	Read-only
FE42h	2 Bytes		Reserved	
FE44h	Doubleword		FS Base (see note 1)	
FE48h	Quadword		Descriptor in memory format	
FE50h	Word	GS	Selector	Read-only
FE52h	2 Bytes		Reserved	
FE54h	Doubleword		GS Base (see note 1)	
FE58h	Quadword		Descriptor in memory format	
FE60h	4 Bytes	GDTR	Reserved	Read-only
FE64h	Word		Limit	
FE66h	2 Bytes		Reserved	
FE68h	Quadword		Descriptor in memory format	

Table 6: SMM Save State

Offset	Size	Contents		Access
FE70h	Word	LDTR	Selector	Read-only
FE72h	Word		Attributes	
FE74h	Doubleword		Limit	
FE78h	Quadword		Base	
FE80h	4 Bytes	IDTR	Reserved	Read-only
FE84h	Word		Limit	
FEB6h	2 Bytes		Reserved	
FE88h	Quadword		Base	
FE90h	Word	TR	Selector	Read-only
FE92h	Word		Attributes	
FE94h	Doubleword		Limit	
FE98h	Quadword		Base	
FEA0h	Quadword	IO_RESTART_RIP		Read-only
FEA8h	Quadword	IO_RESTART_RCX		
FEB0h	Quadword	IO_RESTART_RSI		
FEB8h	Quadword	IO_RESTART_RDI		
FEC0h	Doubleword	SMMFEC0 [SMM IO Trap Offset]		Read-only
FEC4	Doubleword	SMMFEC4 [Local SMI Status]		Read-only
FEC8h	Byte	SMMFEC8 [SMM IO Restart Byte]		Read-write
FEC9h	Byte	SMMFEC9 [Auto Halt Restart Offset]		Read-write
FECAh	Byte	SMMFECA [NMI Mask]		Read-write
FECBh	5 Bytes	Reserved		
FED0h	Quadword	EFER		Read-only
FED8h	Quadword	SVM State		Read-only
FEE0h	Quadword	Guest VMCB physical address		Read-only
FEE8h	Quadword	SVM Virtual Interrupt Control		Read-only
FEF0h	16 Bytes	Reserved		
FEFCh	Doubleword	SMMFEFC [SMM-Revision Identifier]		Read-only
FF00h	Doubleword	SMMFF00 [SMM Base Address (SMM_BASE)]		Read-write
FF04h	28 Bytes	Reserved		
FF20h	Quadword	Guest PAT		Read-only
FF28h	Quadword	Host EFER		
FF30h	Quadword	Host CR4		
FF38h	Quadword	Host CR3		
FF40h	Quadword	Host Cr0		
FF48h	Quadword	CR4		
FF50h	Quadword	CR3		
FF58h	Quadword	CR0		

Table 6: SMM Save State

Offset	Size	Contents	Access
FF60h	Quadword	DR7	Read-only
FF68h	Quadword	DR6	
FF70h	Quadword	RFLAGS	Read-write
FF78h	Quadword	RIP	Read-write
FF80h	Quadword	R15	
FF88h	Quadword	R14	
FF90h	Quadword	R13	
FF98h	Quadword	R12	
FFA0h	Quadword	R11	
FFA8h	Quadword	R10	
FFB0h	Quadword	R9	
FFB8h	Quadword	R8	
FFC0h	Quadword	RDI	Read-write
FFC8h	Quadword	RSI	
FFD0h	Quadword	RBP	
FFD8h	Quadword	RSP	
FFE0h	Quadword	RBX	
FFE8h	Quadword	RDX	
FFF0h	Quadword	RCX	
FFF8h	Quadword	RAX	

Note 1: All addresses are stored in canonical form.

The SMI save state includes most of the integer execution unit. Not included in the save state are: the floating point state, MSRs, and CR2. In order to be used by the SMI handler, these must be saved and restored. The save state is the same, regardless of the operating mode (32-bit or 64-bit).

The following are some offsets in the SMM save state area. The mnemonic for each offset is in the form SMMxxxx, where xxxx is the offset in the save state.

SMMFEC0 SMM IO Trap Offset

Read-only; updated-by-hardware.

If the assertion of SMI is recognized on the boundary of an IO instruction, SMMFEC0 [SMM IO Trap Offset] contains information about that IO instruction. For example, if an IO access targets an unavailable device, the system can assert SMI and trap the IO instruction. SMMFEC0 then provides the SMI handler with information about the IO instruction that caused the trap. After the SMI handler takes the appropriate action, it can reconstruct and then re-execute the IO instruction from SMM. Or, more likely, it can use SMMFEC8 [SMM IO Restart Byte], to cause the core to re-execute the IO instruction immediately after resuming from SMM.

Bits	Description
31:16	**Port: trapped IO port address**. This provides the address of the IO instruction.

15:12	**BPR: IO breakpoint match**.
11	**TF: EFLAGS TF value**.
10:7	Reserved.
6	**SZ32: size 32 bits**. 1=Port access was 32 bits.
5	**SZ16: size 16 bits**. 1= Port access was 16 bits.
4	**SZ8: size 8 bits**. 1=Port access was 8 bits.
3	**REP: repeated port access**.
2	**STR: string-based port access**.
1	**V: IO trap word valid**. 1=The core entered SMM on an IO instruction boundary; all information in this offset is valid. 0=The other fields of this offset are not valid.
0	**RW: port access type**. 0=IO write (OUT instruction). 1=IO read (IN instruction).

SMMFEC4 Local SMI Status

This offset stores status bits associated with SMI sources local to the core. For each of these bits, 1=The associated mechanism generated an SMI.

Bits	Description
31:18	Reserved.
17	**SmiSrcLvtExt: SMI source LVT extended entry**. Read-write. This bit is associated with the SMI sources specified in APIC[530:500] [Extended Interrupt [3:0] Local Vector Table].
16	**SmiSrcLvtLcy: SMI source LVT legacy entry**. Read-write. This bit is associated with the SMI sources specified by the non-extended LVT entries of the APIC.
15:9	Reserved.
8	**MceRedirSts: machine check exception redirection status**. Read-write. This bit is associated with the SMI source specified in MSRC001_0022 [Machine Check Exception Redirection][RedirSmiEn].
7:4	Reserved.
3:0	**IoTrapSts: IO trap status**. Read-write. Each of these bits is associated with each of the respective SMI sources specified in MSRC001_00[53:50] [IO Trap (SMI_ON_IO_TRAP_[3:0])].

SMMFEC8 SMM IO Restart Byte

00h on entry into SMM.

If the core entered SMM on an IO instruction boundary, the SMI handler may write this to FFh. This causes the core to re-execute the trapped IO instruction immediately after resuming from SMM. The SMI handler should only write to this byte if SMMFEC0[V]=1; otherwise, the behavior is undefined.

If a second SMI is asserted while a valid IO instruction is trapped by the first SMI handler, the CPU services the second SMI prior to re-executing the trapped IO instruction. SMMFEC0[V]=0 during the second entry into SMM, and the second SMI handler must not rewrite this byte.

If there is a simultaneous SMI IO instruction trap and debug breakpoint trap, the processor first responds to the SMI and postpones recognizing the debug exception until after resuming from SMM. If debug registers other than DR6 and DR7 are used while in SMM, they must be saved and restored by the SMI handler. If SMMFEC8

[SMM IO Restart Byte], is set to FFh when the RSM instruction is executed, the debug trap does not occur until after the IO instruction is re-executed.

Bits	Description
7:0	**RST: SMM IO Restart Byte**. Read-write.

SMMFEC9 Auto Halt Restart Offset

Bits	Description
7:1	Reserved.
0	**HLT: halt restart**. Read-write. Upon SMM entry, this bit indicates whether SMM was entered from the halt state. 0=Entered SMM on a normal x86 instruction boundary. 1=Entered SMM from the halt state. Before returning from SMM, this bit can be written by the SMI handler to specify whether the return from SMM should take the processor back to the halt state or to the instruction execution state specified by the SMM state save area (normally, the instruction after the halt). 0=Return to the instruction specified in the SMM save state. 1=Return to the halt state. If the return from SMM takes the processor back to the halt state, the HLT instruction is not re-fetched and re-executed. However, the halt special bus cycle is broadcast and the processor enters the halt state.

SMMFECA NMI Mask

Bits	Description
7:1	Reserved.
0	**NmiMask**. Read-write. Specifies whether NMI was masked upon entry to SMM. 0=NMI not masked. 1=NMI masked.

SMMFED8 SMM SVM State

This offset stores the SVM state of the processor upon entry into SMM.

Bits	Description
63:4	Reserved.
3	**HostEflagsIf**: host Eflags IF. Read-only; updated-by-hardware.
2:0	**SvmState**. Read-only; updated-by-hardware. Bits — Definition 000b — SMM entered from a non-guest state. 001b — Reserved. 010b — SMM entered from a guest state. 101b-011b — Reserved. 110b — SMM entered from a guest state with nested paging enabled. 111b — Reserved.

SMMFEFC SMM-Revision Identifier

Read-only; updated-by-hardware. SMM entry state: 0003_0064h

Bits	Description
31:18	Reserved.
17	**BRL**. Base relocation supported.
16	**IOTrap**. IO trap supported.
15:0	**Revision**.

SMMFF00 SMM Base Address (SMM_BASE)

Bits	Description
31:0	See: MSRC001_0111[31:0]. This offset is loaded with the contents of MSRC001_0111.

2.4.6.2.6 Exceptions and Interrupts in SMM

When SMM is entered, the CPU masks INTR, NMI, SMI, INIT, and A20M interrupts. The CPU clears the IF flag to disable INTR interrupts. To enable INTR interrupts within SMM, the SMM handler must set the IF flag to 1. A20M is disabled so that address bit 20 is never masked when in SMM.

Generating an INTR interrupt can be used for unmasking NMI interrupts in SMM. The CPU recognizes the assertion of NMI within SMM immediately after the completion of an IRET instruction. Once NMI is recognized within SMM, NMI recognition remains enabled until SMM is exited, at which point NMI masking is restored to the state it was in before entering SMM.

While in SMM, the CPU responds to the DBREQ and STPCLK interrupts, as well as to all exceptions that may be caused by the SMI handler.

2.4.6.2.7 The Protected ASeg and TSeg Areas

These ranges are controlled by MSRC001_0112 and MSRC001_0113.

2.4.6.2.8 SMM Special Cycles

Special cycles can be initiated on entry and exit from SMM to acknowledge to the system that these transitions are occurring. These are controlled by MSRC001_0015[SMISPCYCDIS, RSMSPCYCDIS].

2.4.6.2.9 Locking SMM

The SMM registers (MSRC001_0112 and MSRC001_0113) can be locked from being altered by setting MSRC001_0015[SmmLock]. BIOS can lock the SMM registers after initialization to prevent unexpected changes to these registers.

2.4.7 Secure Virtual Machine Mode (SVM)

Support for SVM mode is indicated by CPUID Fn8000_0001_ECX[SVM]. If SVM is supported, the DEV registers from D18F3xF0 to D18F3xF8 are visible.

2.4.7.1 BIOS support for SVM Disable

BIOS should include the following user setup options to enable and disable AMD Virtualization™ technology.

- Enable AMD Virtualization™.
  - MSRC001_0114[Svm_Disable] = 0.
  - MSRC001_0114[Lock] = 1.
  - MSRC001_0118[SvmLockKey] = 0000_0000_0000_0000h.
- Disable AMD Virtualization™.
  - MSRC001_0114[Svm_Disable]=1.
  - MSRC001_0114[Lock]=1.
  - MSRC001_0118[SvmLockKey] = 0000_0000_0000_0000h.

BIOS may also include the following user setup option to disable AMD Virtualization™.

- Disable AMD Virtualization™, with a user supplied key.
  - MSRC001_0114[Svm_Disable]=1.
  - MSRC001_0114[Lock]=1.
  - MSRC001_0118[SvmLockKey] programmed with value supplied by user. This value should be stored in NVRAM.

2.4.8 CPUID Instruction

The CPUID instruction provides data about the features supported by the processor. See 3.19 [CPUID Instruction Registers].

2.4.8.1 Multi-Core Support

There are two methods for determining multi-core support. A recommended mechanism is provided and a legacy method is also available for existing operating systems. System software should use the correct architectural mechanism to detect the number of cores by observing CPUID Fn8000_0008_ECX[NC]. The legacy method utilizes the CPUID Fn0000_0001_EBX[LogicalProcessorCount].

2.5 Power Management

The processor supports many power management features in a variety of systems. Table 7 provides a summary of ACPI states and power management features and indicates whether they are supported.

Table 7: Power management support

ACPI/Power Management State	Supported	Description
G0/S0/C0: Working	Yes	
G0/S0/C0: Core P-state transitions	Yes	2.5.3.1 [Core P-states]
G0/S0/C0: NB P-state transitions	Yes	2.5.4.2 [NB Clock Ramping]
G0/S0/C0: Hardware thermal control (HTC)	Yes	2.10.3.1 [PROCHOT_L and Hardware Thermal Control (HTC)]
G0/S0/C0: Thermal clock throttling (SMC controlled)	No	
G0/S0: Low power C-states	Yes	2.5.3.2 [C-states] and 2.5.1.4.2 [Alternate Low Power Voltages]
G1/S1: Stand By (Powered On Suspend)	No	
G1/S3: Stand By (Suspend to RAM)	Yes	2.5.6.1.1 [ACPI Suspend to RAM State (S3)]
G1/S4, S5: Hibernate (Suspend to Disk), Shut Down (Soft Off)	Yes	

Table 7: Power management support

ACPI/Power Management State	Supported	Description
G3 Mechanical Off	Yes	
Parallel VID Interface	No	2.5.1 [Processor Power Planes And Voltage Control]
Serial VID Interface	Yes	
Dual-plane systems	Yes	

2.5.1 Processor Power Planes And Voltage Control

Refer to the Electrical Data Sheet for AMD Family 14h Models 00h-0Fh Processors, #44446 for power plane electrical requirements and definitions.

2.5.1.1 Internal VID Registers

The registers within the processor that contain VID fields all use 7-bit VID encodings. See AMD Voltage Regulator Specification, #40182.

- The VID for VDDCR_NB is controlled by D18F3xDC[NbPs0Vid], D18F6x90[NbPs1Vid], and the internal GPU. See 2.5.4.1 [NB P-states].
- The VID for VDDCR_CPU is controlled by MSRC001_00[6B:64][CpuVid] of the core in the highest-performance P-state.

2.5.1.1.1 VID Encodings

The VID encoding to voltage translations, for all VID codes, are defined by the AMD Voltage Regulator Specification, #40182.

The boot VID is 1.0 V.

2.5.1.1.2 MinVid and MaxVid Check

Hardware limits the minimum and maximum VID code that is sent to the voltage regulator. The allowed limits of MinVid and MaxVid are provided in MSRC001_0071. Prior to generating VID-change commands to SVI, the processor filters the InputVid value to the OutputVid as follows (higher VID codes correspond to lower voltages and lower VID codes correspond to higher voltages):

- If InputVid < MaxVid, OutputVid = MaxVid.
  - Else if (InputVid > MinVid) & (MinVid != 00h), OutputVid = MinVid.
  - Else OutputVid = InputVid.

This filtering is applied regardless of the source of the VID-change command.

2.5.1.2 Serial VID Interface

The processor includes an interface, intended to control external voltage regulators, called the serial VID interface (SVI). Refer to the AMD Voltage Regulator Specification, #40182 for details. The frequency of the SVC is controlled by D18F3xA0[SviHighFreqSel].

2.5.1.3 BIOS Requirements for Power Plane Initialization

1. Initialize D18F3xD8[VSRampSlamTime].
2. Configure D18F3xA0[PsiVidEn, PsiVid] and D18F3x128[NbPsiVidEn, NbPsiVid]. See 2.5.1.4.1 [PSI_L

Bit].
3. Program D18F3xDC[NbPs0Vid] = FCRxFE00_6000[NbPs0Vid] - 1.
4. Program D18F3xDC[NbPs0Vid] = FCRxFE00_6000[NbPs0Vid].
5. Request the lowest valid voltage specified by D18F3x15C using D0F0x64_x6A and D0F0x64_x6B. See 2.5.1.5.2 [Software-Initiated Voltage Transitions].
6. Perform 2.9.3.7 [DRAM Training].
7. After 2.9.3.7 [DRAM Training] is complete:
   - If (GpuEnabled), BIOS requests the highest valid voltage specified by D18F3x15C. BIOS makes this request using GMMx770 and GMMx774 prior to video BIOS initialization. See 2.5.1.5.2 [Software-Initiated Voltage Transitions]. After initialization, the graphics driver may reduce the voltage based on the display configuration.

After performing the above steps, BIOS may request VDDCR_CPU and/or VDDCR_NB voltage changes at any time.

2.5.1.4 Low Power Features

2.5.1.4.1 PSI_L Bit

The processor supports additional system power savings through the use of a low-voltage state indicator, the PSI_L bit. The PSI_L bit in the data field of the SVI command can be used by the voltage regulator to place itself into a more power efficient mode. The PSI_L bit can be controlled independently for the VDDCR_CPU and VDDCR_NB planes. The PSI_L bit is enabled through D18F3xA0[PsiVidEn] and D18F3x128[NbPsiVidEn]. Once enabled, the state of the PSI_L bit is controlled by D18F3xA0[PsiVid] and D18F3x128[NbPsiVid]. PSI_L bit changes only occur on VID changes.

2.5.1.4.1.1 BIOS Requirements for PSI_L

Enabling PSI_L for the VDDCR_CPU and VDDCR_NB planes depends on the voltage regulator and is therefore system specific. Depending on the voltage regulator being used, AMD recommends one of the following methods:

- PSI_L always clear:
  - To clear PSI_L for the VDDCR_CPU plane, program D18F3xA0[PsiVidEn]=1 and D18F3xA0[PsiVid]=0. To clear PSI_L for the VDDCR_NB plane, program D18F3x128[NbPsiVidEn]=1 and D18F3x128[NbPsiVid]=0.

- PSI_L always set:
  - To set PSI_L for the VDDCR_CPU plane, program D18F3xA0[PsiVidEn]=0. To set PSI_L for the VDDCR_NB plane, program D18F3x128[NbPsiVidEn]=0.

- PSI_L set/clear based on current requirements (VDDCR_CPU only):
  - If the voltage regulator requires that PSI_L be set or cleared at a certain current level, BIOS uses the following pseudo-code:

```
psi_vrm_current = current at which the regulator allows PSI_L;
psi_inrush_current = inrush current during a voltage transition;

for (each valid P-state starting with P0) {
    pstate_current = MSRC001_00[6B:64][IddDiv, IddVal];
    pstate_voltage = MSRC001_00[6B:64][CpuVid];
```

```
        if (current P-state is P0) {
            prev_voltage = 7Fh;
        } else {
            prev_voltage = MSRC001_00[6B:64][CpuVid] of previous P-state;
        }

        if (current P-state is last valid P-state) {
            next_pstate_current = 0;
        } else {
            next_pstate_current = psi_inrush_current +
            MSRC001_00[6B:64][IddDiv, IddVal] of the next P-state;
        }

        if ((pstate_current <= psi_vrm_current) &&
            (next_pstate_current <= psi_vrm_current) &&
            (pstate_voltage != prev_voltage)){
            program D18F3xA0[PsiVid] = pstate_voltage;
            program D18F3xA0[PsiVidEn] = 1;
            break;
        }
    }
```

Please contact your voltage regulator vendor and your AMD representative to determine the best method.

2.5.1.4.2 Alternate Low Power Voltages

In order to save power, voltages lower than those normally used may be applied to the VDDCR_CPU or VDDCR_NB power planes while the processor is in a C-state or the GPU is idle. These lower voltages are defined as follows:

- C6Vid (D18F3x128[C6Vid]): C6Vid specifies a VDDCR_CPU voltage that does not retain the CPU caches or the cores' microarchitectural state, nor allows for execution. As a result, hardware flushes caches and saves the cores' microarchitectural state to DRAM before transitioning to C6Vid. See 2.5.3.2.3.4 [Package C6 (PC6) State].
- GnbIdleAdjustVid (FCRxFE00_7070[GnbIdleAdjustVid]): GnbIdleAdjustVid specifies a voltage offset that is subtracted from the VDDCR_NB voltage. The resulting voltage allows all components on VDDCR_NB to function if the GPU is clock or power gated and incapable of causing voltage transients due to changes in activity. GnbIdleAdjustVid is enabled using FCRxFF30_0191[GfxIdleVoltChgMode, GfxIdleVoltChgEn]. GnbIdleAdjustVid is not applied when the internal GPU is disabled (see GpuEnabled in 1.4 [Definitions]).

2.5.1.4.3 Power Gating

The processor can remove power from an individual core. This is referred to as power gating. Gating power to a subcomponent causes its internal microarchitectural state and, if applicable, any data in its caches to be lost. When entering a power gated state, hardware saves any needed data, either internally or to DRAM, and flushes caches. When exiting a power gated state, hardware performs any required resets and restores any needed data. See 2.5.3.2.3.2 [Core C6 (CC6) State].

2.5.1.5 Voltage Transitions

The processor supports dynamic voltage transitions on the VDDCR_CPU and VDDCR_NB planes. These transitions are controlled by hardware. VDDCR_CPU and VDDCR_NB voltage levels may be transitioned during state changes involving reset, boot, P-state changes, C-state changes, and stop-grant. In all cases, the voltage is *slammed*; this means that the VID code passed to the voltage regulator changes from the old value to the new value without stepping through intermediate values. The voltage regulator ramps the voltage directly from the starting voltage to the final voltage. See the AMD Voltage Regulator Specification, #40182.

If a voltage increase is requested, the processor waits the amount of time specified by D18F3xD8[VSRampSlamTime] before sending any additional voltage change requests to the voltage regulator or before beginning a frequency transition. If a voltage decrease is requested, the processor does not wait any time before sending additional voltage changes or beginning frequency changes. The processor continues execution of code during voltage changes when in the C0 state.

2.5.1.5.1 Hardware-Initiated Voltage Transitions

When software requests any of the following power state changes, or hardware determines that any of the following state changes are necessary, hardware coordinates the necessary voltage changes:

- VDDCR_CPU:
  - Core P-state transition. See 2.5.3.1 [Core P-states].
  - Package C-state transition. See 2.5.3.2 [C-states].
  - S-state transition. See 2.5.6.1 [S-states].

- VDDCR_NB:
  - NB P-state transition. See 2.5.4.1 [NB P-states].
  - S-state transition. See 2.5.6.1 [S-states].
  - GPU state transitions.

2.5.1.5.2 Software-Initiated Voltage Transitions

Software can request voltage changes on the VDDCR_NB power plane using the following control/status register pairs:

- D0F0x64_x6A and D0F0x64_x6B
- GMMx770 and GMMx774

The voltage requests from each register pair are considered independently by hardware when taking voltage plane dependencies into account (see 2.5.2.2 [Dependencies Between Subcomponents on VDDCR_NB]). To make a voltage change request, software uses the following sequence:

1. Ensure VoltageChangeEn==1 in the control register. If software needs to program VoltageChangeEn=1, software must perform this register write independently of the writes in the following steps.
2. Program VoltageLevel to the desired voltage and toggle VoltageChangeReq in the control register.
3. The voltage change is complete when VoltageChangeReq in the control register is equal to VoltageChangeAck in the status register.

Software can force a VDDCR_NB voltage change using D0F0x64_x6A and GMMx770. To do so, software programs VoltageForceEn=1 in the respective register before toggling VoltageChangeReq when making a voltage change request. If this is done, the voltage requested overrides any other VDDCR_NB voltage requests

made by software when determining voltage plane dependencies (see 2.5.2.2 [Dependencies Between Subcomponents on VDDCR_NB]). NB P-state transitions still request voltage transitions as normal when software forces a voltage change using this mechanism. If software forces a voltage change using both D0F0x64_x6A and GMMx770, the voltage requested in D0F0x64_x6A takes precedence.

D18F3xDC[NbPs0Vid] also causes VDDCR_NB voltage transitions when in NBP0. See 2.5.4.1 [NB P-states].

2.5.2 Frequency and Voltage Domain Dependencies

2.5.2.1 Dependencies Between Cores

Whenever a P-state or C-state is requested on a core (see 2.5.3.1 [Core P-states] and 2.5.3.2 [C-states]), hardware must take frequency and voltage domain dependencies into account when determining whether to make the requested state change. Whenever software requests different power management states for the cores on a multi-core processor, the processor decides which state to target based on a set of policy controls, described below.

- If multiple cores request different P-states, frequency changes independently on each core based on the P-state requested by that core. The VDDCR_CPU voltage is determined by the highest-performance P-state requested on any core. See 2.5.3.1 [Core P-states].
- If some cores request non-C0 C-states while other cores are in C0, core C-state actions are taken independently by the cores requesting those states. Package C-state actions are not taken. The VDDCR_CPU voltage is determined by the highest-performance P-state requested on any core.
- If all cores request non-C0 C-states, core C-state actions are taken independently by each core and package C-state actions are determined by the shallowest C-state request made by any core.

2.5.2.2 Dependencies Between Subcomponents on VDDCR_NB

Many subcomponents of the processor including the NB, the GPU, and the root complex reside on the VDDCR_NB power plane. Hardware must take voltage domain dependencies into account when determining whether to make a voltage change requested by one of the subcomponents. Whenever a state transition occurs that causes a voltage change request (see 2.5.1.5.1 [Hardware-Initiated Voltage Transitions]), or software makes a voltage change request (see 2.5.1.5.2 [Software-Initiated Voltage Transitions]), the VDDCR_NB voltage requested by the processor is determined by the highest voltage (lowest VID) request made by any of the subcomponents or by software. In addition, software can force the GNB component of VDDCR_NB voltage changes. See 2.5.1.5.2 [Software-Initiated Voltage Transitions].

2.5.3 CPU Power Management

2.5.3.1 Core P-states

Core P-states are operational performance states characterized by a unique combination of core frequency and voltage. The processor supports up to 8 core P-states, specified in MSRC001_00[6B:64]. Out of reset, the voltage and frequency of the cores is specified by MSRC001_0071[StartupPstate].

Support for dynamic core P-state changes is indicated by more than one enabled selection in MSRC001_00[6B:64][PstateEn]. The DID and VID for each core P-state is specified in MSRC001_00[6B:64]. All DID parameters for equivalent P-states must be programmed to equivalent values for all cores. For examples, P0 on core0 must have the same DID value as P0 on core1, P1 on core0 must have the same DID value as P1 on core1, and so on. Refer to MSRC001_00[6B:64] for further details on programming requirements. The COF for core P-states is a function of the main PLL frequency and the DID. See D18F3xD4[MainPllOpFreqId] for more details on the main PLL frequency and MSRC001_00[6B:64][CpuDidLSD] for more details on the

DID.

Software requests core P-state changes for each core independently using the hardware P-state control mechanism (also known as “fire and forget”). Support for hardware P-state control is indicated by CPUID Fn8000_0007_EDX[HwPstate]=1b. P-state transitions using the hardware P-state control mechanism are not allowed until the P-state initialization requirements defined in 2.5.3.1.5 [BIOS Requirements for Core P-State Initialization and Transitions] are complete.

2.5.3.1.1 Core P-state Control

Core P-states are dynamically controlled by software and are exposed through ACPI objects (see 2.5.3.1.7 [ACPI Processor P-State Objects]). Software requests a core P-state change by writing a 3 bit index corresponding to the desired P-state number to MSRC001_0062 [P-State Control] of the appropriate core. For example, to request P3 for core 0 software would write 011b to core 0’s MSRC001_0062[PstateCmd].

Hardware sequences the frequency and voltage changes necessary to complete a P-state transition as specified by 2.5.3.1.4 [Core P-state Transition Behavior] with no additional software interaction required. Hardware also coordinates frequency and voltage changes when differing P-state requests are made on cores that share a frequency or voltage plane. See 2.5.2 [Frequency and Voltage Domain Dependencies] for details about hardware coordination.

Core P-states are changed without interacting with an external chipset. However, the chipset is notified of core P-state changes by the P-state special cycle if MSRC001_001F[EnaPStateSpCyc]=1.

2.5.3.1.2 Core P-state Visibility

MSRC001_0063[CurPstate] reflects the number of the current P-state of each core. For example, core 1 MSRC001_0063[CurPstate]=010b indicates core 1 is in the P2 state.

The voltage being provided to a core may not correspond to the VID code specified by the current P-state of the core due to voltage plane dependencies. See 2.5.2 [Frequency and Voltage Domain Dependencies].

2.5.3.1.3 Core P-state Limits

Core P-states may be limited to lower-performance states under certain conditions, including:

- HTC. See D18F3x64 [Hardware Thermal Control (HTC)][HtcPstateLimit].
- PROCHOT_L assertion. See 2.10.3.1 [PROCHOT_L and Hardware Thermal Control (HTC)].
- SB-TSI. See 2.10.2 [Sideband Temperature Sensor Interface (SB-TSI)].

The current core P-state limit is provided in MSRC001_0061[CurPstateLimit]. In addition, the maximum P-state value (lowest-performance P-state), regardless of the source, is limited as specified in MSRC001_0061[PstateMaxVal].

2.5.3.1.4 Core P-state Transition Behavior

The following rules specify how P-states changes function and interact with other system or processor states:

- If the P-state number is increasing (the core is moving to a lower-performance state), then the COF is changed first, followed by the VID change. If the P-state number is decreasing, then the VID is changed first followed by the COF.
- See 2.5.1.5 [Voltage Transitions] for details about voltage transitions made during P-state changes.

- If multiple commands are issued that affect the P-state of a domain prior to when the processor initiates the change of the P-state of that domain, then the processor operates on the last one issued.
- Once a P-state change starts, the P-state state machine (PSSM) continues through completion unless interrupted by a RESET_L de-assertion. If multiple P-state changes are requested concurrently, the PSSM may group the associated VID changes separately from the associated COF changes.
- If RESET_L asserts, all cores are transitioned to C0 and to the P-state specified by MSRC001_0071[StartupPstate].
  - After a warm reset MSRC001_0062 and MSRC001_0063 are consistent with MSRC001_0071[CurP-state].
- The OS controls the P-state through MSRC001_0062, independent of the P-state limits described in MSRC001_0061[PstateMaxVal, CurPstateLimit]. P-state limits interact with OS-directed P-state transitions as follows:
  - The P-state for a core is changed by hardware to MSRC001_0061[PstateMaxVal] if MSRC001_0061[PstateMaxVal] changes and the current P-state number is higher than (lower-performance than) MSRC001_0061[PstateMaxVal].
  - The P-state for a core is changed by hardware to MSRC001_0061[CurPstateLimit] if MSRC001_0061[CurPstateLimit] changes and the current P-state number is lower than (higher-performance than) MSRC001_0061[CurPstateLimit].
- See 2.5.3.2.7 [C-state initiated P-state Changes] for information regarding P-state usage while cores are in C-states and after cores exit C-states.

2.5.3.1.5 BIOS Requirements for Core P-State Initialization and Transitions

1. Check that CPUID Fn8000_0007_EDX[HwPstate]=1. If not, P-states are not supported, no P-state related ACPI objects should be generated, and BIOS must skip the rest of these steps.
2. Complete the requirements in 2.5.1.3 [BIOS Requirements for Power Plane Initialization].
3. Determine the valid set of P-states based on the enabled P-states indicated in MSRC001_00[6B:64] [P-State [7:0]][PstateEn].
4. Optionally perform 2.5.3.1.6 [Processor and System Board Power Delivery Check].
5. If only one P-state is enabled in MSRC001_00[6B:64] [P-State [7:0]][PstateEn], then BIOS must not generate ACPI-defined P-state objects described in 2.5.3.1.7 [ACPI Processor P-State Objects]. Otherwise, the ACPI objects should be generated to enable P-state support.

2.5.3.1.6 Processor and System Board Power Delivery Check

BIOS may disable core P-states that require higher power delivery than the system board can support. This power delivery compatibility check is designed to prevent system failures caused by exceeding the power delivery capability of the system board for the VDDCR_CPU power plane. BIOS can optionally notify the user if P-states are detected that exceed the system board power delivery capability. Modifications to MSRC001_00[6B:64] must be applied equally to all cores. This check does not ensure functionality for all processor/system board combinations.

MSRC001_00[6B:64][PstateEn] must be set to 0 for any P-state MSR where PstateEn=1 and the processor current requirement (ProcIddMax), defined by the following equation, is greater than the system board current delivery capability.

```
ProcIddMax = MSRC001_00[6B:64][IddValue] * 1/10^MSRC001_00[6B:64][IddDiv] *
    (D18F3xE8[CmpCap]+1);
```

The power delivery check should be applied starting with P0 and continue with increasing P-state indices (1, 2, 3, and 4) for all enabled P-states. Once a compatible P-state is found using the ProcIddMax equation, the check

is complete. All enabled P-states with higher indices are defined to be lower power and performance, and are therefore compatible with the system board.

Example:
- MSRC001_0065[IddValue] = 32d.
- MSRC001_0065[IddDiv] = 0d.
- D18F3xE8[CmpCap] = 1d.
- ProcIddMax = 32 * 1 * 2 = 64 A.

In this example, the system board must be able to supply >= 64 A on VDDCR_CPU in order to support P1 for this processor. If the system board current delivery capability is < 64 A per plane then BIOS must clear MSRC001_0065[PstateEn] to 0 for all cores on this processor node, and continue by checking P2 in the same fashion.

If no P-states are disabled on the processor while performing the power delivery compatibility check then BIOS does not need to take any action.

If at least one P-state is disabled on the processor by performing the power delivery compatibility check and at least one P-state remains enabled for the processor, then BIOS must perform the following steps:
1. Ensure D18F3xDC[PstateMaxVal] is programmed to the BIOS recommended value.
2. If the P-state indicated by MSRC001_0063[CurPstate] is disabled by the power delivery compatibility check, then BIOS must request a transition to an enabled P-state using MSRC001_0062[PstateCmd] and wait for MSRC001_0063[CurPstate] to reflect the new value.
3. Copy the contents of the enabled P-state MSRs (MSRC001_00[6B:64]) to the highest performance P-state locations and disable any duplicate P-states. E.g. if P0 and P1 are disabled by the power delivery compatibility check and P2 through P4 remain enabled, then the contents of MSRC001_0066 through MSRC001_0068 should be copied to MSRC001_0064 through MSRC001_0066 and MSRC001_00[68:67][PstateEn] should be cleared to 0.
4. Request a P-state transition to the P-state MSR containing the COF/VID values currently applied. E.g. If MSRC001_0063[CurPstate]=100b and P4 P-state MSR information is copied to P2 in step 2, then BIOS should write 010b to MSRC001_0062[PstateCmd] and wait for MSRC001_0063[CurPstate] to reflect the new value.
5. Adjust the P-state parameters affected by the P-state MSR copy:
   - If the P-state indicated by D18F3x64[HtcPstateLimit] is enabled, subtract the number of P-states that are disabled by the power delivery compatibility check from the following fields:
     - D18F3x64[HtcPstateLimit]
     - D18F3xDC[PstateMaxVal]

     For example, if P0 and P1 are disabled, then each of the above register fields would have 2 subtracted from them. This calculation must not wrap; the subtraction must saturate at 0.
   - If the P-state indicated by D18F3x64[HtcPstateLimit] is disabled, do not change D18F3x64[HtcPstateLimit] or D18F3xDC[PstateMaxVal].

If the processor has all P-states disabled after performing the power delivery compatibility check, then BIOS must perform the following steps. Note that this does not ensure operation, and that BIOS should notify the user of the incompatibility between the processor and system board if possible.
1. Ensure D18F3xDC[PstateMaxVal] is programmed to the BIOS recommended value.
2. If MSRC001_0063[CurPstate] is not pointing to the highest number (lowest performance) enabled P-state:
   A. Write the highest number (lowest performance) enabled P-state to MSRC001_0062[PstateCmd].
   B. Wait for MSRC001_0063[CurPstate] to reflect the new value.
3. If MSRC001_0063[CurPstate]!=000b:
   A. Copy the contents of the P-state MSR pointed to by MSRC001_0063[CurPstate] to MSRC001_0064.

B. Write MSRC001_0064[PstateEn]=1.
C. Write MSRC001_0062[PstateCmd]=000b.
D. Wait for MSRC001_0063[CurPstate] to reflect the new value.

4. Adjust the P-state parameters affected by the P-state MSR copy:
   - If the P-state indicated by D18F3x64[HtcPstateLimit] is enabled, program the following fields to 0:
     - D18F3x64[HtcPstateLimit]
     - D18F3xDC[PstateMaxVal]
   - If the P-state indicated by D18F3x64[HtcPstateLimit] is disabled, do not change D18F3x64[HtcPstateLimit] or D18F3xDC[PstateMaxVal].
5. Disable P1 through P7 by clearing PstateEn in MSRC001_0065 through MSRC001_006B to 0.

2.5.3.1.7 ACPI Processor P-State Objects

ACPI 2.0 and ACPI 3.0 processor performance control for processors reporting CPUID Fn8000_0007_EDX[HwPstate]=1 is implemented through two objects whose presence indicates to the OS that the platform and CPU are capable of supporting multiple performance states. Processor performance states are not supported with ACPI 1.0b. BIOS must provide the _PCT object, _PSS object, and define other ACPI parameters to support operating systems that provide native support for processor P-state transitions. Other optional ACPI objects are also described in the following sections.

The following rules apply to BIOS generated ACPI objects in multi-core systems. Refer to the appropriate ACPI specification (http://www.acpi.info) for additional details:

- All cores must expose the same number of performance states to the OS.
- The respective performance states displayed to the OS for each core must have identical performance and power-consumption parameters (e.g. P0 on core 0 must have the same performance and power-consumption parameters as P0 on core1, P1 on core 0 must have the same parameters as P1 on core 1, however P0 can be different than P1)
- Performance state objects must be present under each processor object in the system.

2.5.3.1.7.1 _PCT (Performance Control)

BIOS must declare the performance control object parameters as functional fixed hardware. This definition indicates the processor driver understands the architectural definition of the P-state interface associated with CPUID Fn8000_0007_EDX[HwPstate]=1.

- Perf_Ctrl_Register = Functional Fixed Hardware
- Perf_Status_Register = Functional Fixed Hardware

2.5.3.1.7.2 _PSS (Performance Supported States)

A unique _PSS entry is created for each P-state. The value contained in the _PSS Control field is written to MSRC001_0062 [P-State Control] to request a P-state change to the CoreFreq of the associated _PSS object. The value contained in MSRC001_0063 [P-State Status] can be used to identify the _PSS object of the current P-state by equating MSRC001_0063[CurPstate] to the value of the _PSS Status field. See 2.5.3.1 [Core P-states].

BIOS must loop through each of MSRC001_00[6B:64] [P-State [7:0]] applying the following formulas to create the fields for the _PSS objects.

- CoreFreq (MHz) = Calculated using the COF formula documented in MSRC001_00[6B:64][CpuDidLSD].

- Power (mW)
  - Convert MSRC001_00[6B:64][CpuVid] to a voltage by referring to the AMD Voltage Regulator Specification, #40182.
  - Power (mW) = voltage * MSRC001_00[6B:64][IddValue] * 1/10^MSRC001_00[6B:64][IddDiv] * 1000
- TransitionLatency (us) = BusMasterLatency (us) = 0 us.
- Control/Status
  - The highest-performance P-state must have the _PSS control and status fields programmed to 0.
  - Each lower-performance P-state must have the _PSS control and status fields programmed in ascending order.

The following example will show the _PSS entries for a hypothetical scenario. Given a processor configured with these register values:
- D18F3xD4[MainPllOpFreqId]=8
- The P-state MSRs have the following values:
  - MSRC001_0064=8000_0028_0000_4800h
  - MSRC001_0065=8000_0023_0000_5802h
  - MSRC001_0066=8000_001C_0000_6810h
  - MSRC001_0067=8000_01BC_0000_7812h
  - MSRC001_00[6B:68]=0

The _PSS objects in this example would have the following values:
- P0
  - CoreFreq=2400MHz
  - Power=40W
  - TransitionLatency=BusMasterLatency=0
  - Control=Status=0
- P1
  - CoreFreq=1600MHz
  - Power=35W
  - TransitionLatency=BusMasterLatency=0
  - Control=Status=1
- P2
  - CoreFreq=1200MHz
  - Power=25W
  - TransitionLatency=BusMasterLatency=0
  - Control=Status=2
- P3
  - CoreFreq=960MHz
  - Power=15W
  - TransitionLatency=BusMasterLatency=0
  - Control=Status=2

In this example, no other P-states would be defined.

2.5.3.1.7.3 _PPC (Performance Present Capabilities)

The _PPC object is optional. Refer to the ACPI specification for details on use and content.

2.5.3.1.7.4 _PSD (P-State Dependency)

An ACPI 3.0 _PSD object is required for each core:

- NumberOfEntries = 5.
- Revision = 0.
- Domain = 0.
- CoordType = FCh. (SW_ALL)
- NumProcessors = CPUID Fn8000_0008_ECX[NC]+1.

2.5.3.1.7.5 Fixed ACPI Description Table (FADT) Entries

Declare the PSTATE_CNT entry as 00h.

2.5.3.1.8 XPSS (Microsoft® Extended PSS) Object

Some Microsoft® operating systems require an XPSS object to make P-state changes function properly. A BIOS that implements an XPSS object has special requirements for the _PCT object. See the Microsoft *Extended PSS ACPI Method Specification* for the detailed requirements to implement these objects.

2.5.3.2 C-states

C-states are processor power states. C0 is the operational state in which instructions are executed. All other C-states are low-power states in which instructions are not executed.

2.5.3.2.1 C-state Names and Numbers

C-states are often referred to by an alphanumeric naming convention, C1, C2, C3, etc. The mapping between ACPI defined C-states and AMD specified C-state actions is not direct. The actions taken by the processor when entering a low-power C-state are specified by D18F4x118 and D18F4x11C and are configured by software. See 2.5.3.2.3 [C-state Actions] for information about AMD specific actions.

2.5.3.2.2 C-state Request Interface

C-states are dynamically requested by software and are exposed through ACPI objects (see 2.5.3.2.8 [ACPI Processor C-state Objects]). C-states can be requested on a per-core basis. Software requests a C-state change in one of two ways, either by executing the HLT instruction or by reading from an IO address specified by MSRC001_0073[CstateAddr] plus an offset of 0 through 7 (see D18F4x118 and D18F4x11C for details). Execution of the HLT instruction is equivalent to reading from the IO address specified by MSRC001_0073[CstateAddr]. The processor always returns 0 for this IO read. When software requests a C-state transition:

1. Hardware determines which C-state actions were requested. See 2.5.3.2.3 [C-state Actions].
2. Hardware evaluates any frequency and voltage domain dependencies. See 2.5.2 [Frequency and Voltage Domain Dependencies].
3. Hardware evaluates any enabled and unmasked C-state request monitors. See 2.5.3.2.4 [C-state Request Monitors].
4. Hardware enters the deepest C-state that is allowed.

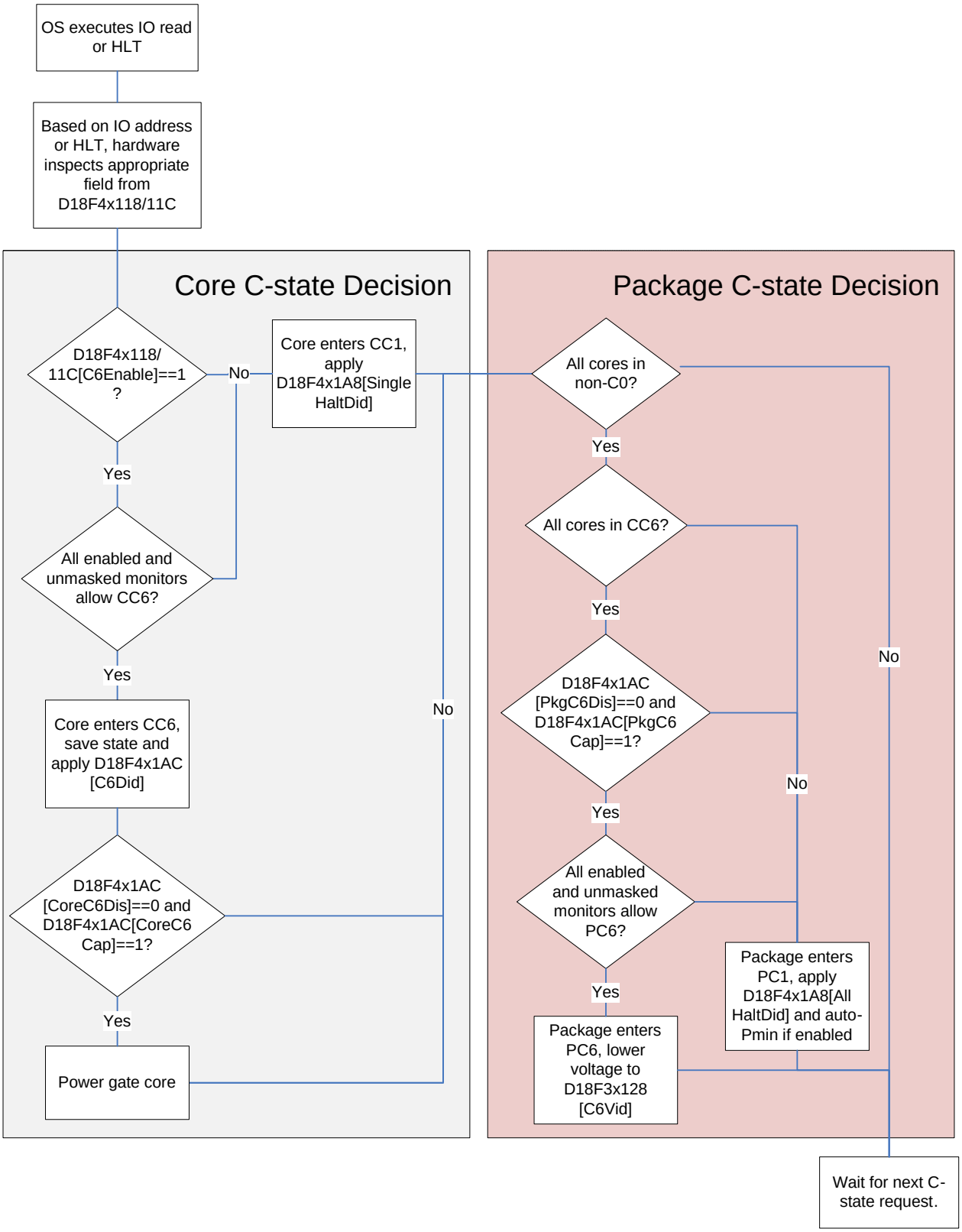


Figure 3: C-state entry flowchart

2.5.3.2.3 C-state Actions

Each C-state has per-core and per-package functionality. See 2.5.2.1 [Dependencies Between Cores]

Each C-state action is enabled using D18F4x118 and D18F4x11C. Multiple actions can be enabled for a given IO address. If no actions are specified when a core enters a core C-state, or if hardware determines that no actions are acceptable, the core enters the CC1 state. If no actions are specified when the package enters a package C-state, or if hardware determines that no actions are acceptable, the package enters the PC1 state.

2.5.3.2.3.1 Core C1 (CC1) State

When a core enters the CC1 state, its clock ramps down to the frequency specified by D18F4x1A8[SingleHaltCpuDid].

2.5.3.2.3.2 Core C6 (CC6) State

When a core enters the CC6 state, it executes the following sequence:

1. L1 and L2 caches are flushed to DRAM by hardware.
2. Internal core state is saved to DRAM by hardware.
3. The core clock ramps down to the frequency specified by D18F4x1AC[C6Did].
4. Power is removed from the core if possible as specified by D18F4x1AC[CoreC6Cap] and D18F4x1AC[CoreC6Dis].

The events which cause a core to exit the CC6 state are specified in 2.5.3.2.6 [Exiting C-states].

If a warm reset occurs while a core is in CC6, all MCA registers in the core shown in Table 36 are cleared to 0. See 2.16 [Machine Check Architecture].

2.5.3.2.3.3 Package C1 (PC1) State

The processor enters the PC1 state with auto-Pmin when all of the following are true:

- All cores are in the CC1 state or deeper.

If D18F4x1AC[CstPminEn] indicates that auto-Pmin is enabled when the processor enters PC1, the P-state for all cores is transitioned as specified by 2.5.3.2.7.1 [Auto-Pmin]. Regardless of the state of D18F4x1AC[CstPminEn], all core clocks are ramped to the frequency specified by D18F4x1A8[AllHaltCpuDid].

2.5.3.2.3.4 Package C6 (PC6) State

The processor enters the PC6 state when all of the following are true:

- All cores enter the CC6 state.
- The C-state action field targeted by each core's C-state request has the C6Enable bit programmed to indicated entry into PC6 is allowed. See D18F4x118 and D18F4x11C.
- PC6 is supported and enabled as specified by D18F4x1AC[PkgC6Cap] and D18F4x1AC[PkgC6Dis].

When the package enters PC6, VDDCR_CPU is transitioned to the VID specified by D18F3x128[C6Vid].

2.5.3.2.4 C-state Request Monitors

Deeper C-states have higher entry and exit latencies but provide greater power savings than shallower C-states. To help balance the performance and power needs of the system, the processor can limit access to specific C-states in certain scenarios.

2.5.3.2.4.1 DMA Tracking

DMA activity can indicate that entering deep C-states may hamper performance. DMA activity tracking allows DMA activity to limit access to certain C-states.

2.5.3.2.4.1.1 DMA Activity Tracking

The processor tracks DMA activity and can disallow access to or PC6 as a result. When a package C-state transition request is made, the processor denies access to the respective C-state if any DMA traffic specified by D18F4x120[DeepCstDMATrackEn] has occurred within the hysteresis time specified by D18F4x120[CstDMATrackHyst].

2.5.3.2.4.2 FCH Messaging

The FCH can be notified when the processor transitions package C-states. See D18F4x120[CstateMsgDis]. If the processor sends a message when entering PC6, the FCH sends a return message notifying the processor whether the C-state is allowed. If the C-state is disallowed, the processor enters PC1. The processor can ignore the FCH response as specified by D18F4x120[DeepCstAllowMsgEn].

If the FCH does not respond to the processor within the time specified by D18F4x120[FchTO], a timeout occurs. This causes the processor to take the actions specified by D18F4x120[DeepCstTOPol].

2.5.3.2.4.3 Interrupt Monitors

The processor supports two mechanisms to track interrupt behavior, a timer tick monitor and an interrupt rate monitor. A short timer tick period or a large number of interrupts can indicate that entering deep C-states may hamper performance. Both monitors can control access to CC6 or PC6 independently.

2.5.3.2.4.3.1 Timer Tick Monitor

The timer tick monitor tracks interrupts that are delivered from the FCH on a regular interval, generally referred to as timer tick interrupts. When software changes the period of the timer tick interrupt, the FCH reports the new period to the processor.

The timer tick monitor operates in either interval mode or duration mode as specified by D18F4x124[TrackTimerTickInterEn]. The mode determines the time frame used by the timer tick monitor to make decisions. In interval mode, the monitor uses the timer tick period. In duration mode, the monitor calculates the time remaining until the next expected timer tick interrupt whenever a C-state transition request is made. The monitor compares the time frame specified by the mode to thresholds defined by D18F4x124[IntMonCC6Lmt] and D18F4x124[IntMonPkgC6Lmt]. If the time frame is less than or equal to the threshold, access to the appropriate C-state is disallowed.

The timer tick monitor is enabled using D18F4x124[IntMonCC6En] and D18F4x124[IntMonPkgC6En].

The timer tick monitor cannot be used to track periodic local APIC timer interrupts.

2.5.3.2.4.3.2 Interrupt Rate Monitor

The interrupt rate monitor tracks the behavior of all interrupts in the system using independent counters for CC6 and PC6. The counters behave as follows:

- On a regular interval specified by D18F4x134[IntRateCC6BurstLen] and D18F4x134[IntRatePkgC6BurstLen], the processor determines if any interrupts were received during the interval. If so, it increments the appropriate counter by 1.
- The processor decrements each counter by 1 at a constant rate, specified by D18F4x134[IntRateCC6DecrRate] and D18F4x134[IntRatePkgC6DecrRate].
- The counters saturate at values specified by D18F4x134[IntRateCC6MaxDepth] and D18F4x134[IntRatePkgC6MaxDepth].

Whenever software requests a C-state transition, the current value in each counter is compared against a threshold specified by D18F4x134[IntRateCC6Threshold] and D18F4x134[IntRatePkgC6Threshold]. If the counter value is greater than the threshold, access to the appropriate C-state is disallowed.

2.5.3.2.4.4 Residency Monitors

Per-core C0 residency and non-C0 residency times are tracked by the processor. Time spent in CC1 can count towards C0 residency time or non-C0 residency time as specified by D18F4x124[MonitorEnableMode]. High C0 residency or low non-C0 residency indicates the system is in use and entering deep C-states may hamper performance. Each residency monitor controls access to CC6 independently.

2.5.3.2.4.4.1 C0 Residency Monitor

The C0 residency monitor uses an internal counter to determine if access to CC6 is allowed. The counter behaves as follows:

- Whenever a core enters a non-C0 C-state, the duration of the core's most recent C0 residency is compared to the threshold specified by D18F4x124[C0MonCC6Lmt]. If the residency was less than the threshold, the counter is incremented, otherwise it is decremented.
- The counter saturates at the value specified by D18F4x124[C0MonCC6Cntr].

When a core attempts to enter CC6, the transition is allowed if the counter is saturated at the value specified by D18F4x124[C0MonCC6Cntr], otherwise access is disallowed.

The C0 residency monitor is enabled using D18F4x124[C0MonCC6En].

2.5.3.2.4.4.2 Non-C0 residency Monitor

When a core attempts to enter CC6, the transition is allowed if the duration of the core's most recent non-C0 residency was greater than the threshold specified by D18F4x128[NonC0MonCC6Lmt], otherwise the transition is disallowed.

The non-C0 residency monitor is enabled using D18F4x128[NonC0MonCC6En].

2.5.3.2.4.5 C-state Monitor Masking

Each of the interrupt and residency C-state request monitors can be masked on a per-CAF basis, as specified by D18F4x1A4. If a C-state monitor is masked for a given CAF, requests to transition to CC6 using that CAF ignore the results of that C-state monitor.

2.5.3.2.5 C-states and Probe Requests

If a core is in a C-state in which its caches are not flushed, it must service any probe requests that occur. The following algorithm defines the actions taken by the processor prior to and after servicing probe requests from non-C0 C-states:

```
Probe request comes in to a core in a non-C0 C-state and core caches have not been
flushed;

if (the package is in PC1 && D18F4x1AC[CstPminEn]==1) {
    Ramp the core frequency to D18F4x1A8[PService];
    Service the probe request;
    Wait for D18F3xD4[ClkRampHystSel] timer to expire;
    Ramp the core to the frequency specified by the current C-state;
} elseif (a core is in CC1 while other cores are in C0 || the package is in PC1) {
    if ((D18F4x1A8[CpuProbEn]==0) || (core divisor is /16 or deeper)) {
        Ramp the core to the frequency specified by the current P-state;
        Service the probe request;
        Wait for D18F3xD4[ClkRampHystSel] timer to expire;
        Ramp the core to the frequency specified by the current C-state;
    } else {
        Service the probe request;
    }
}
```

2.5.3.2.6 Exiting C-states

The following events may cause the processor to exit a non-C0 C-state and return to C0:

- INTR.
- NMI.
- SMI.
- INIT.
- RESET_L assertion.

If an INTR is received while a core is in a non-C0 C-state, the state of EFLAGS[IF] and the mechanism used to enter the non-C0 C-state determine the actions taken by the processor.

- Entry via HLT, EFLAGS[IF]==0: The interrupt does not wake up the core.
- Entry via HLT, EFLAGS[IF]==1: The interrupt wakes the core and the core begins execution. Any pending interrupts are serviced.
- Entry via IO read, EFLAGS[IF]==0: The interrupt wakes the core and the core begins execution. Any pending interrupts are not serviced until EFLAGS[IF] is programmed to 1.
- Entry via IO read, EFLAGS[IF]==1: The interrupt wakes the core and the core begins execution. Any pending interrupts are serviced.

Regardless of the entry mechanism, if APIC80 indicates the core is at a higher priority level than the INTR received, the core does not wake up.

2.5.3.2.7 C-state initiated P-state Changes

C-state changes can automatically cause core P-state changes. These features make use of a timer called the PService timer and a P-state called the PService state. The PService timer is enabled using

D18F4x1A8[PServiceTmrEn], expires as specified by D18F4x1A8[PServiceTmr], and counts/resets as described below. The PService state is specified by D18F4x1A8[PService].

Any P-state limits in effect are honored when making these P-state changes.

2.5.3.2.7.1 Auto-Pmin

If D18F4x1AC[CstPminEn] indicates auto-Pmin is enabled when the processor enters package C1, hardware automatically transitions all cores to the PService state. In this case, if the PService timer is enabled when the processor exits PC1, the PService timer resets and begins counting down. All cores remain at the PService state until one of the following occurs:

- The PService timer expires while the package is in C0: In this case, the cores transition back to the last P-state requested by software.
- The package enters PC1 with auto-Pmin: In this case, the PService timer stops counting and the cores remain in the PService state. See 2.5.3.2.3.3 [Package C1 (PC1) State].
- The package enters PC6: In this case the PService timer stops counting and the actions associated the requested package C-state occur. See 2.5.3.2.3.4 [Package C6 (PC6) State].

If the PService timer is disabled when the processor exits PC1, the cores transition back to the last P-state requested by software.

2.5.3.2.7.2 Exiting PC6

If the PService timer is enabled when the processor exits PC6, all cores transition to the P-state specified by D18F4x1AC[PstateIdCoreOffExit]. The cores remain in this state until one of the following occurs:

- The PService timer expires while the package is in C0: In this case, the cores transition back to the last P-state requested by software.
- The package enters PC1 without auto-Pmin: In this case, the PService timer continues counting. If it expires while in PC1, the cores remain in the P-state specified by D18F4x1AC[PstateIdCoreOffExit] until they return to C0, at which time they transition to the last P-state requested by software.
- The package enters PC1 with auto-Pmin: In this case, the PService timer stops counting and the cores enter the PService state.
- The package enters PC4 or PC6: In this case the PService timer stops counting and the actions associated the requested package C-state occur. See 2.5.3.2.3.4 [Package C6 (PC6) State].

If the PService timer is disabled when the processor exits PC6, the cores transition back to the last P-state requested by software.

2.5.3.2.8 ACPI Processor C-state Objects

ACPI 2.0 and ACPI 3.0 processor power control is implemented through the _CST object whose presence indicates to the OS that the platform and processor are capable of supporting multiple power states. BIOS must provide the _CST object and define other ACPI parameters to support operating systems that provide native support for processor C-state transitions. Other optional ACPI objects are also described in the following sections.

The _CST object is not supported with ACPI 1.0b. BIOS should provide FADT entries as outlined below to support operating systems that are not ACPI 2.0 capable.

The following rules apply to BIOS generated ACPI objects in multi-core systems. Refer to the appropriate ACPI specification for additional details:

- C-state objects must be present under each processor object in the system.

2.5.3.2.8.1 _CST (C-state)

The _CST objects contains information about each C-state the processor supports. BIOS provides a _CST object for each processor object as follows:

- Count: 1.
- C-state package:
  - Register:
    - AddressSpaceKeyword: SystemIO
    - RegisterBitWidth: 8
    - RegisterBitOffset: 0
    - RegisterAddress: MSRC001_0073[CstateAddr] + 1
    - AccessSize: 1 (byte)
  - Type: 2
  - Latency: 100
  - Power: 0

2.5.3.2.8.2 _CRS

BIOS must declare in the root host bridge _CRS object that the IO address range from MSRC001_0073[CstateAddr] to MSRC001_0073[CstateAddr]+7 is consumed by the host bridge.

2.5.3.2.8.3 Fixed ACPI Description Table (FADT) Entries

Declare the following FADT entries:

- P_LVL2_LAT = 100.
- P_LVL3_LAT = 1001.
- FLAGS.PROC_C1 = 1.
- FLAGS.P_LVL2_UP = 1.

Declare the following P_BLK entries:

- P_LVL2 = MSRC001_0073[CstateAddr] + 1.
- P_LVL3 = 0.

2.5.3.2.9 BIOS Requirements for C-state Initialization

1. Initialize MSRC001_0073[CstateAddr] on each core to a region of the IO address map with 8 consecutive available addresses. The cores are not required to use the same IO addresses.
2. Program D18F4x1A8[PService] to the index of lowest-performance P-state with MSRC001_00[6B:64][PstateEn]==1 on core 0.
3. Program D18F4x1AC[CstPminEn] to 1.
4. If CC6 or PC6 is supported as indicated by D18F4x1AC[CoreC6Cap, PkgC6Cap]:
   - Ensure D18F2x118[C6DramLock] and D18F4x12C[C6Base] are programmed as specified by 2.9.6 [DRAM CC6/PC6 Storage].
   - If PC6 is supported, program D18F4x1AC[PstateIdCoreOffExit] to the index of lowest-performance P-state with MSRC001_00[6B:64][PstateEn]==1 on core 0.
   - Program D18F4x118 to 0000_0101h.
5. Generate ACPI objects as described in 2.5.3.2.8 [ACPI Processor C-state Objects].

2.5.3.3 Effective Frequency

The effective frequency interface allows software to discern the average, or effective, frequency of a given core over a configurable window of time. This provides software a measure of actual performance rather than forcing software to assume the current frequency of the core is the frequency of the last P-state requested. This can be useful when the P-state is being limited by HTC.

The effective frequency can be determined using MSR0000_00E7 [Max Performance Frequency Clock Count (MPERF)] and MSR0000_00E8 [Actual Performance Frequency Clock Count (APERF)] and the following steps:

1. At some point in time, write 0 to both MSRs.
2. At a later point in time, read both MSRs.
3. Effective frequency = (value read from MSR0000_00E8 / value read from MSR0000_00E7) * P0 frequency.

Additional notes:
- The amount of time that elapses between steps 1 and 2 is determined by software.
- It is software's responsibility to disable interrupts or any other events that may occur in between the write of MSR0000_00E7 and the write of MSR0000_00E8 in step 1 or between the read of MSR0000_00E7 and the read of MSR0000_00E8 in step 2.
- The behavior of MSR0000_00E7 and MSR0000_00E8 may be modified by MSRC001_0015[EffFreqCntMwait].
- The effective frequency interface provides +/- 50 MHz accuracy if the following constraints are met:
  - Effective frequency is read no more often than one time per millisecond.
  - When reading or writing MSR0000_00E7 and MSR0000_00E8 software executes only MOV instructions, and no more than 3 MOV instructions, between the two RDMSR or WRMSR instructions.
- MSRC001_0015[TscFreqSel] must be set to 1 for the effective frequency interface to function correctly.

2.5.4 Northbridge Power Management

2.5.4.1 NB P-states

The processor supports dynamic northbridge frequency (NCLK) changes and voltage (VDDCR_NB) change requests, referred to as NB P-states. NB P-states are enabled/disabled with D18F6x90[NbPsCtrlDis]. The northbridge switches between two states, NBP0, a higher-performance higher-power consumption state, and NBP1, a lower-performance lower-power consumption state. The COF and VID controls for each NB P-state are specified by:

- NBP0
  - COF: D18F3xDC[NbPs0NclkDiv].
  - VID: D18F3xDC[NbPs0Vid].
- NBP1
  - COF: D18F6x90[NbPs1NclkDiv].
  - VID: D18F6x90[NbPs1Vid].

No runtime software control is needed for NB P-states. Hardware autonomously transitions the NB P-state when all necessary criteria are met. The following criteria are used by hardware to determine when an NB P-state transitions is necessary:

- Core P-states: Based on the setting of D18F6x94[CpuPstateThrEn], hardware can take the current P-state of each core into account when determining whether to transition NB P-states. If D18F6x94[CpuPstate-ThrEn]=1 and all cores are in a P-state with equal or lesser performance than the P-state specified by D18F6x94[CpuPstateThr], a transition from NBP0 to NBP1 may be triggered. If D18F6x94[CpuPstate-ThrEn]=1 and any core is in a greater performance P-state than specified by D18F6x94[CpuPstateThr], a transition from NBP1 to NBP0 may be triggered.
- Core C-states: When all cores have entered a non-C0 C-state and an amount of time specified by D18F6x94[NbPsNonC0Timer] has elapsed, a transition from NBP0 to NBP1 may be triggered.
- GPU activity: If the internal GPU is enabled, the GPU driver can specify the level of GPU activity that can cause an NB P-state transition. See D18F6x90[NbPs1GnbSlowIgn].
- NB P-state forcing: Software can force an NB P-state transition unconditionally using D18F6x90[NbPs-ForceSel, NbPsForceReq].
- DRAM self-refresh: If DRAM is in self-refresh, no NB P-state changes occur.

Once hardware determines that an NB P-state transition is necessary, the following parameters may delay the NB P-state transition:

- NB P-state residency timer: When a transition from NBP0 to NBP1 occurs, transitions back to NBP0 are prevented until the time specified by D18F6x94[NbPs1ResTmrMin] has elapsed. When a transition from NBP1 to NBP0 occurs, transitions back to NBP1 are prevented until the time specified by D18F6x94[NbPs0ResTmrMin] has elapsed.
- DMA activity: Based on the setting of D18F6x94[NbPs1NoTransOnDma], DMA activity or the assertion of DMAACTIVE_L can prevent NB P-state transitions.

The following diagrams show how all of the criteria are logically combined together.

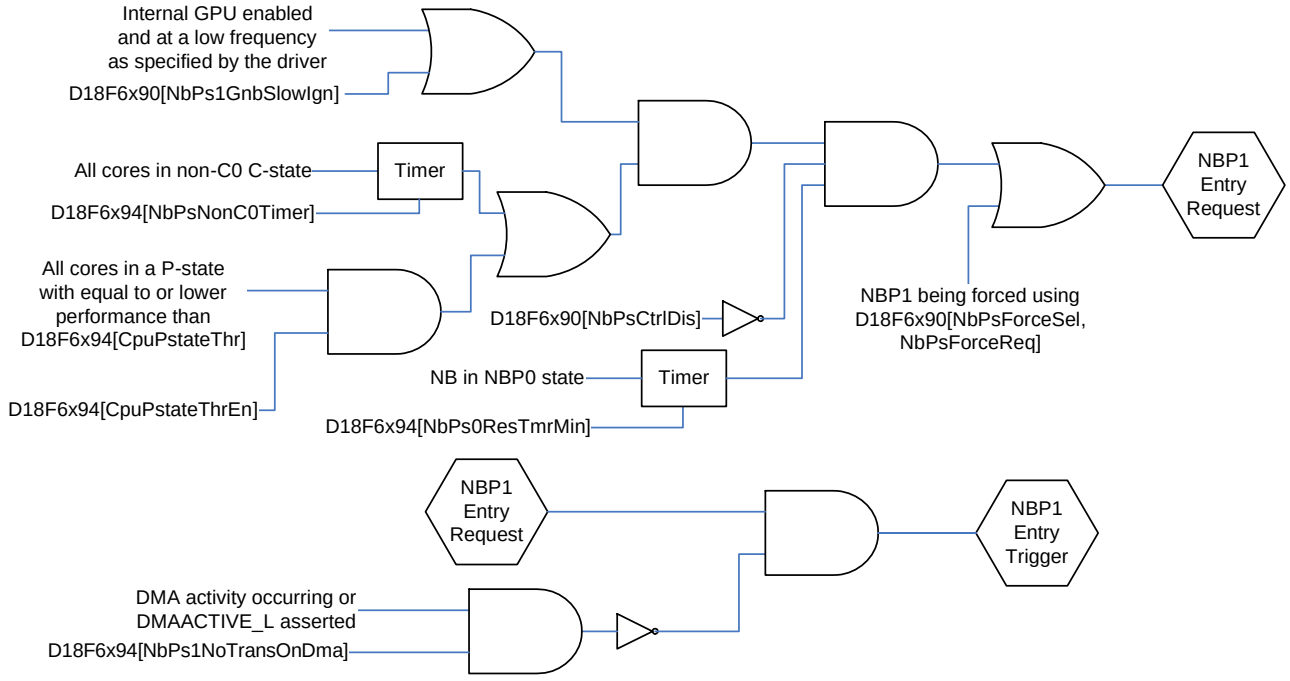


Figure 4: NBP0 to NBP1 transition determination

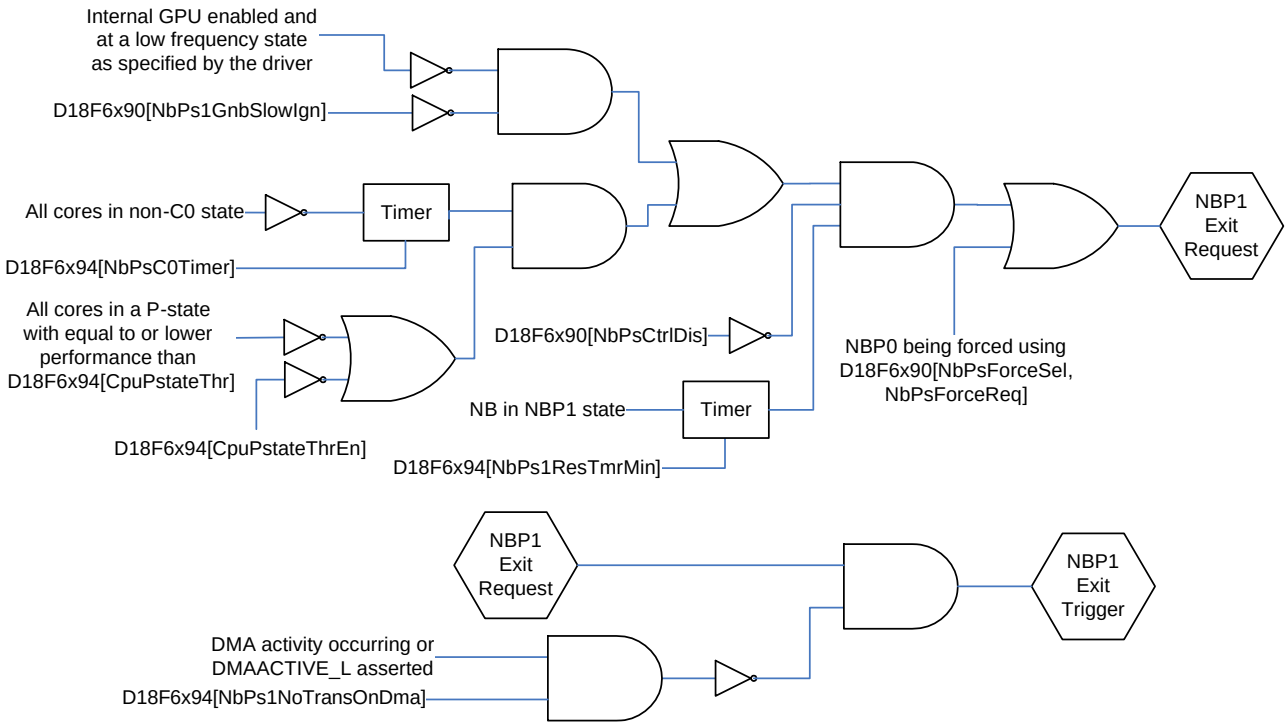


Figure 5: NBP1 to NBP0 transition determination

Once hardware determines that a NB P-state transition is necessary and that it may make the transition, hardware executes the following sequence:

1. Set D18F6x98[NbPsTransInFlight] = 1.
2. If the transition is from NBP1 to NBP0, request a VDDCR_NB transition from D18F6x90[NbPs1Vid] to D18F3xDC[NbPs0Vid] and wait for it to complete (see 2.5.1.5 [Voltage Transitions]).
3. Stop memory traffic and place DRAM into self-refresh.
4. Transition from current NCLK divisor to new NCLK divisor as specified by D18F3xDC[NbPs0NclkDiv] and D18F6x90[NbPs1NclkDiv].
5. Update memory settings.
6. Take DRAM out of self-refresh and allow memory traffic.
7. If the transition is from NBP0 to NBP1, request a VDDCR_NB transition from D18F3xDC[NbPs0Vid] to D18F6x90[NbPs1Vid].

When hardware makes an NB P-state change, D18F2xF4_x30[DbeGskFifoNumerator] and D18F2xF4_x31[DbeGskFifoDenominator] must be programmed to non-zero values or undefined behavior results.

2.5.4.1.1 BIOS Requirements for NB P-state Initialization During DRAM Training

1. Determine the target MEMCLK frequency. See 2.9.3.2.2.1 [Requirements for DRAM Frequency Change During Training].
2. If (target MEMCLK frequency < FCRxFE00_7009[NbPs0NclkDiv] frequency):
   - Program D18F3xDC[NbPs0NclkDiv] to the minimum divisor that generates a 50% duty cycle clock where (target MEMCLK frequency >= (D18F3xD4[MainPllOpFreqId] frequency) / divisor).

   Else:
   - Program D18F3xDC[NbPs0NclkDiv] = FCRxFE00_7009[NbPs0NclkDiv].

3. Wait for D18F3xDC[NclkFreqDone] == 1.
4. If (D18F6x90[NbPsCap]):
   - If (target MEMCLK frequency/2 > FCRxFE00_7006[NbPs1NclkDiv] frequency):
     - Program D18F6x90[NbPs1NclkDiv] to the maximum divisor that generates a 50% duty cycle clock where (target MEMCLK frequency/2 <= (D18F3xD4[MainPllOpFreqId] frequency) / divisor).
     - If (D18F6x90[NbPs1NclkDiv] frequency) <= (FCRxFE00_7006[MaxNbFreqAtMinVid] frequency) program D18F6x90[NbPs1Vid] = FCRxFE00_6002[NbPs1VidAddl].
     - If (D18F6x90[NbPs1NclkDiv] frequency) > (FCRxFE00_7006[MaxNbFreqAtMinVid] frequency) program D18F6x90[NbPs1Vid] = FCRxFE00_6002[NbPs1VidHigh].
   - Else:
     - Program D18F6x90[NbPs1NclkDiv] = FCRxFE00_7006[NbPs1NclkDiv].
     - Program D18F6x90[NbPs1Vid] = FCRxFE00_6000[NbPs1Vid].
5. If ((D18F6x90[NbPsCap]) && (D18F3xDC[NbPs0NclkDiv] frequency < D18F6x90[NbPs1NclkDiv] frequency)):
   - Program D18F6x90[NbPs1NclkDiv] = D18F3xDC[NbPs0NclkDiv].

See 2.9.3.4.7 [NB P-states for DCT/DRAM Initialization and Training].

2.5.4.1.2 System BIOS Requirements for NB P-state Operation During POST

In addition to 2.5.4.1.1 [BIOS Requirements for NB P-state Initialization During DRAM Training], BIOS creates a data structure in memory containing information about the processor for use by the GPU driver. Please see your AMD representative for more information.

2.5.4.1.3 Software Controlled NB P-states

Software may use the hardware NB P-state mechanism to force an NB P-state change. This is enabled as specified by D18F6x90[NbPsForceReq]. Once enabled, software selects the current NB P-state by programming D18F6x90[NbPsForceSel]. Whenever software requests an NB P-state transition, it occurs as soon as there are no other NB P-state transitions in flight (D18F6x98[NbPsTransInFlight]) and no DMA activity. The transition occurs regardless of all other criteria listed in 2.5.4.1 [NB P-states]. Software can determine when the transition is complete by reading D18F6x98[NbPs1Act].

2.5.4.2 NB Clock Ramping

NCLK is opportunistically ramped down whenever DRAM enters self-refresh and the package is in a C-state as specified by D18F6x9C[NclkRedSelfRefrAlways]. NCLK is ramped down to a divisor specified by D18F6x9C[NclkRedDiv].

When DRAM exits self-refresh NCLK ramps back up to the divisor specified by the current NB P-state (see 2.5.4.1 [NB P-states]). This occurs either serially or in parallel with DDR PHY DLL re-lock. See D18F6x9C[NclkRampWithDllRelock].

2.5.4.3 NB Clock Gating

Portions of NCLK can be gated at certain times. This is enabled using D18F3xDC[NbClockGateEn] and D18F3xD4[DisNclkGatingIdle]. The following describes each portion of NCLK that can be gated and the associated controls.

- IFQ: NCLK distributed to the IFQ is gated when all of the following are true:
  - No traffic through the NB.
  - All cores are in a non-C0 C-state and the IFQ has been empty for the hysteresis time specified by

D18F3xDC[NbClockGateHyst]. See 2.5.3.2.3 [C-state Actions].

- DRAM: NCLK distributed to the DRAM is gated when all of the following are true:
  - DRAM NCLK gating is enabled as specified by D18F3xD4[ClockGatingEnDram].
  - IFQ clock gating is active.
  - DRAM is in self-refresh. See 2.5.5.1 [DRAM Self-Refresh].

- NBCIF: NCLK distributed to the NBCIF of each core is gated when any of the following are true:
  - NBCIF gating is enabled as specified by D18F3xDC[CnbCifClockGateEn].
  - IFQ gating is active.
  - The core is in CC6. See 2.5.3.2.3 [C-state Actions].

2.5.5 DRAM Power Management

2.5.5.1 DRAM Self-Refresh

DRAM is placed into self-refresh in the following three scenarios:
- While in S3 as specified by the DramSr bit in D18F3x80 and D18F3x84.
- While in S0, due to NB P-state transitions (see 2.5.4.1 [NB P-states]).
- While in S0, due to stutter mode (see 2.5.5.1.1 [Stutter Mode]).

2.5.5.1.1 Stutter Mode

DRAM is most commonly placed in self-refresh due to stutter mode. The display buffer in the GPU is a combination of a large buffer known as the DMIF (Display Memory Interface FIFO) and a smaller line buffer. The DMIF takes data originating from DRAM, and sends it to the line buffer to draw to the screen. When the data level in the DMIF is full, DRAM is placed in self-refresh, and incoming DRAM requests are queued. As the DMIF drains, it eventually falls below a predefined watermark level, at which point hardware pulls DRAM out of self-refresh and services all the requests in the queue. Once all the requests are complete and the DMIF is full again, a transition back into self-refresh occurs if the stutter mode conditions are still met.

The following requirements must be met before hardware places DRAM into self-refresh:
- D18F4x1A8[DramSrEn]==1.
- All cores are in a non-C0 C-state.
- One of the following is true:
  - The GPU is idle and the internal display buffer is full.
  - The internal GPU is disabled.
- There is no pending traffic from the link.

Once the above requirements are met:
- If D18F4x1A8[DramSrHystEnable]==0, hardware places DRAM into self-refresh immediately.
- If D18F4x1A8[DramSrHystEnable]==1, hardware waits for the time specified by D18F4x1A8[DramSrHyst]. If any of the above requirements are violated during that time, hardware aborts the attempt to enter self-refresh, resets the timer, and attempts to enter self-refresh when the requirements are met. If the time specified by D18F4x1A8[DramSrHyst] expires without any of the above requirements being violated, hardware places DRAM into self-refresh.

Once DRAM is in self-refresh, hardware removes it from self-refresh whenever any of the above requirements are no longer met.

To save additional power, hardware can tri-state MEMCLK and shut down the DDR phy DLL when DRAM is

placed in self-refresh. See D18F4x1A8[MemTriStateEn]. If D18F2x90[DisDllShutdownSR]==0, D18F4x1A8[MemTriStateEn], D18F3x84[Smaf4DramMemClkTri], D18F3x84[Smaf6DramMemClkTri] must be programmed to 1.

2.5.5.1.1.1 System BIOS Requirements for Stutter Mode Operation During POST

BIOS creates a data structure in memory containing information about the processor for use by the GPU driver. Please see your AMD representative for more information.

2.5.5.2 M_EVENT_L

M_EVENT_L is a level sensitive input to the processor. When asserted, the actions specified by D18F2xA4 are taken. M_EVENT_L is generally asserted to indicate that a DRAM high temperature condition exists. The minimum assertion time for M_EVENT_L is 15 ns. The minimum de-assertion time for M_EVENT_L is 15 ns.

- M_EVENT_L is pulled to VDDIO_MEM_S on the motherboard.
- M_EVENT_L is ignored while:
  - PWROK is de-asserted.
  - RESET_L is asserted.
- BIOS must ensure that throttling is disabled (D18F2xA4[ThrottleEn[1:0]]=00b) until DRAM training is complete.
- See 2.9.7 [DRAM On DIMM Thermal Management].

2.5.6 System Power Management

2.5.6.1 S-states

S-states are ACPI defined sleep states. S0 is the operational state. All other S-states are low-power states in which the various voltage rails in the system may or may not be powered. See the ACPI specification for descriptions of each S-state.

2.5.6.1.1 ACPI Suspend to RAM State (S3)

The processor supports the ACPI-defined S3 state. Software is responsible for restoring the state of the processor's registers when resuming from S3. All registers in the processor that BIOS initialized during the initial boot must be restored. The method used to restore the registers is system specific.

During S3 entry, system memory enters self-refresh mode. Software is responsible for bringing memory out of self-refresh mode when resuming from S3. To bring memory out of self-refresh mode, see 2.9.3 [DCT/DRAM Initialization and Resume].

Many of the system board power planes for the processor are powered down during S3. Refer to the Electrical Data Sheet for AMD Family 14h Models 00h-0Fh Processors, #44446 for the following:

- Power plane electrical requirements during S3.
- Power plane sequencing requirements on S3 entry and exit.
- System signal states for both inputs (e.g. PWROK and RESET_L) and outputs (e.g. VID[*], PSI_L bit, THERMTRIP_L, etc.) during S3.
- System signal sequencing requirements on S3 entry and exit.
- System management message sequencing on S3 entry and exit.

2.6 Performance Monitoring

The processor includes support for two methods of monitoring processor performance: performance monitor counters and instruction based sampling (IBS).

2.6.1 Performance Monitor Counters

The performance monitor counters are used by software to count specific events that occur in the processor. MSRC001_00[03:00] and MSRC001_00[07:04] specify the events to be monitored and how they are monitored. All of the events are specified in 3.24 [Performance Counter Events].

2.6.2 Instruction Based Sampling (IBS)

IBS is a code profiling mechanism that enables the processor to select a random instruction fetch or micro-op after a programmed time interval has expired and record specific performance information about the operation. An interrupt is generated when the operation is complete as specified by MSRC001_103A [IBS Control]. An interrupt handler can then read the performance information that was logged for the operation.

The IBS mechanism is split into two parts: instruction fetch performance controlled through MSRC001_1030 [IBS Fetch Control (IbsFetchCtl)]; and instruction execution performance controlled through MSRC001_1033 [IBS Execution Control (IbsOpCtl)]. Instruction fetch sampling provides information about instruction TLB and instruction cache behavior for fetched instructions. Instruction execution sampling provides information about micro-op execution behavior. The data collected for instruction fetch performance is different from the data collected for instruction execution performance. Support for the IBS feature is indicated by the CPUID Fn8000_0001_ECX[IBS].

Instruction fetch performance is profiled by recording the following performance information (see MSRC001_1030, MSRC001_1031, and MSRC001_1032 for details of the events) for the tagged instruction fetch:

- If the instruction fetch completed or was aborted.
- The number of clock cycles spent on the instruction fetched.
- If the instruction fetch hit or missed the instruction cache.
- If the instruction fetch hit or missed the L1 and L2 TLBs.
- The linear and physical address associated with the fetch.

Instruction execution performance is profiled by tagging one micro-op. Instructions that decode to more than one micro-op return different performance data depending upon which micro-op associated with the instruction is tagged. The following performance information (see MSRC001_1033, MSRC001_1034, MSRC001_1035, MSRC001_1036, MSRC001_1037, MSRC001_1038 and MSRC001_1039 for details of the events) is returned for the tagged micro-op:

- Branch status for branch micro-ops.
- The number clocks from when the micro-op was tagged until the micro-op retires.
- The number clocks from when the micro-op completes execution until the micro-op retires.
- Source information for DRAM, MMIO and IO access.
- If the operation was a load or store that missed the data cache.
- If the operation was a load or store that hit or missed the L1 and L2 TLBs.
- The linear and physical address associated with a load or store operation.

2.7 Configuration Space

PCI-defined configuration space was originally defined to allow up to 256 bytes of register space for each function of each device; these first 256 bytes are called base configuration space (BCS). It was expanded to support up to 4096 bytes per function; bytes 256 through 4095 are called extended configuration space (ECS). The processor includes configuration space registers located in both BCS and ECS.

Configuration space is accessed by the processor through two methods:
- IO-space configuration: IO instructions to addresses CF8h and CFCh.
  - Enabled through IOCF8 [IO-Space Configuration Address][ConfigEn], which allows access to BCS.
  - Access to ECS enabled through MSRC001_001F [Northbridge Configuration (NB_CFG)][EnableCf8ExtCfg].
  - Use of IO-space configuration can be programmed to generate GP faults through MSRC001_0015 [Hardware Configuration (HWCR)][IoCfgGpFault].
  - SMI trapping for these accesses is specified by MSRC001_0054 [IO Trap Control (SMI_ON_IO_TRAP_CTL_STS)] and MSRC001_00[53:50] [IO Trap (SMI_ON_IO_TRAP_[3:0])].
- MMIO configuration: configuration space is a region of memory space.
  - The base address and size of this range is specified by MSRC001_0058 [MMIO Configuration Base Address]. The size is controlled by the number of configuration-space bus numbers supported by the system. Accesses to this range are converted configuration space as follows:
    - Address[31:0] = {0000b, bus[7:0], device[4:0], function[2:0], offset[11:0]}.

2.7.1 MMIO Configuration Coding Requirements

MMIO configuration space is normally specified to be the uncacheable (UC) memory type. Instructions used to read MMIO configuration space are required to take the following form:

```
mov eax/ax/al, <any_address_mode>;
```

Instructions used to write MMIO configuration space are required to take the following form:

```
mov <any_address_mode>, eax/ax/al;
```

No other source/target registers may be use other than eax/ax/al.

In addition, all such accesses are required not to cross any naturally aligned doubleword boundary. Access to MMIO configuration space registers that do not meet these requirements result in undefined behavior.

2.7.2 MMIO Configuration Ordering

Since MMIO configuration cycles are not serializing in the way that IO configuration cycles are, their ordering rules relative to posted may result in unexpected behavior.

Therefore, processor MMIO configuration space is designed to match the following ordering relationship that exists naturally with IO-space configuration: if a CPU generates a configuration cycle followed by a posted-write cycle, then the posted write is held in the processor until the configuration cycle completes. As a result, any unexpected behavior that might have resulted if the posted-write cycle were to pass MMIO configuration cycle is avoided.

2.7.3 Processor Configuration Space

The processor includes configuration space as described in 3 [Registers]. Accesses to unimplemented registers of implemented functions are ignored: writes dropped; reads return 0's. Accesses to unimplemented functions

also ignored: writes are dropped; however, reads return all F's. The processor does not log any master abort events for accesses to unimplemented registers or functions.

Accesses to device numbers of devices not implemented in the processor are routed based on the configuration map registers. If such requests are master aborted, then the processor can log the event.

2.8 The Northbridge (NB)

The processor includes a single northbridge that provides the interfaces to the core(s), system memory, and system IO devices. The northbridge includes all power planes except VDDCR_CPU; see 2.5.1 [Processor Power Planes And Voltage Control].

The northbridge is responsible for routing transactions sourced from cores and link to the appropriate core, cache, DRAM, or link. See 2.4.4 [System Address Map].

2.8.1 Northbridge (NB) Architecture

The three major northbridge blocks are the northbridge front end (FRE), northbridge cross bar (XBAR) and the DRAM controller (DCT). The FRE interfaces with the core(s). The DCT maintains cache coherency and maintains a queue of incoming requests. The XBAR is a switch that routes packets between the FRE, the DCT, and the link.

2.8.2 Northbridge Buffer Allocation Recommendations

Table 8. Recommended buffer settings

Condition	D18F3x6C					D18F3x74					D18F3x7C				D18F3x17C	
	UpHiNpreqDBC	UpHiPreqDBC	UpLoRespDBC	UpLoNpreqDBC	UpLoPreqDBC	UpHiNpreqCBC	UpHiPreqCBC	UpLoRespCBC	UpLoNpreqCBC	UpLoPreqCBC	FreePoolBC	LoPriNPBC	LoPriPBC	CpuBC	HiPriNPBC	HiPriPBC
D0F0x98_x1E[HiPriEn]==0	0h	0h	1h	1h	Eh	0h	0h	8h	9h	7h	19h	1h	1h	1h	0h	0h
D0F0x98_x1E[HiPriEn]==1	1h	0h	1h	1h	Dh	1h	0h	8h	8h	7h	18h	1h	1h	1h	1h	0h

2.8.3 DMA Exclusion Vectors (DEV)

The DEV is a set of protection tables in system memory that inhibit IO accesses to ranges of system memory. The tables specify link-defined UnitIDs or RequesterIDs (Bus, Device, Function) that are allowed access to physical memory space on a 4 KB page basis. Multiple protection domains are supported, each with independent DEV tables and supported UnitIDs/RequesterIDs. See D18F3xF0 [DEV Capability Header].

2.8.4 Northbridge Routing

2.8.4.1 Address Space Routing

There are four main types of address space routed by the NB: (1) memory space targeting system DRAM, (2) memory space targeting IO (MMIO), (3) IO space, and (4) configuration space. The NB routing registers are accessed through function 1, offsets 40 through F4.

2.8.4.1.1 DRAM and MMIO Memory Space

For memory-space transactions, the physical address, cacheability type, access type, and DRAM/MMIO destination type (see 2.4.4.1.2 [Determining The Access Destination for CPU Accesses]) are presented to the NB for further processing as follows:

- IO-device accesses are processed as follows:
  - If the access matches D18F1x[B8,B0,A8,A0,98,90,88,80] [Memory Mapped IO Base], then the transaction is routed to the root complex;
  - Else, if the access matches D18F1x40 [DRAM Base], then the access is routed to the DCT;
  - Else, the access is routed to the UMI.
- For core accesses the routing is determined based on the DRAM/MMIO destination:
  - If the destination is DRAM:
    - If the access matches D18F1x40, then the transaction is routed to the DCT;
    - Else, the access is routed to the UMI.
  - If the destination is MMIO:
    - If the access matches D18F1x[B8,B0,A8,A0,98,90,88,80], then the transaction is routed to the root complex;
    - Else, the access is routed to the UMI.

2.8.4.1.2 IO Space

IO-space transactions from the link or cores are routed as follows:

- If the access matches D18F1xC0 [IO-Space Base], then the transaction is routed to root complex;
- Else, the access is routed to the UMI.

2.8.5 Physical Address Space

The processor supports 36 address bits of coherent memory space (64 gigabytes) as indicated by CPUID Fn8000_0008_EAX [Long Mode Address Size Identifiers]. The processor master aborts link requests with non-zero physical address bits [63:36].

2.9 DRAM Controller (DCT)

The processor includes one DRAM controller (DCT). The DCT controls one 64-bit DDR3 DRAM channel. A DRAM channel consists of the group of DRAM interface pins connecting to one series of DIMMs.

The DCT operates on physical addresses translated into normalized addresses corresponding to the values programmed into D18F2x[4C:40] [DRAM CS Base Address]. Normalized addresses only include address bits within a DCT's range. The physical to normalized address translation varies based on memory hoisting settings. See 2.9.5 [Memory Hoisting].

The following restrictions limit the DIMM types and configurations supported by the DCTs:

- All DIMMs connected to a processor are required to operate at the same MEMCLK frequency.
- Registered DIMMs are not supported.
- x4 (by 4) DIMMs are not supported.
- Quad rank DIMMs are not supported.

The tables below list the maximum DIMM speeds supported by the processor for different configurations. The motherboard should comply with the FT1 Processor Motherboard Design Guide (MBDG) to achieve the rated speeds. In cases where MBDG design options exist, lower-quality options may compromise the maximum achievable speed; motherboard designers should assess the tradeoffs.

Table 9: DCT Definitions

Term	Definition
DdrRate	The DDR data rate (MT/s)
DIMM0	DIMM slots 0-1. The DIMMs are numbered from 0 to 1 where DIMM0 is the DIMM closest to the processor and DIMM1 is the DIMM farthest from the processor.
DIMM1	
DR	Dual Rank
NP	No DIMM populated
NumDimmSlots	The number of motherboard DIMM slots per channel
SO-DIMM	Small outline dual in-line memory module.
SR	Single Rank
UDIMM	Unbuffered dual in-line memory module.
VDDIO_MEM_S	DDR power supply in V

Table 10: DDR3 UDIMM Maximum Frequency Support for FT1

DIMM Slots/Ch	DIMMs	DIMMs		Frequency[1] (MT/s)
		SR	DR	1.5V
1	1	1	-	1066
		-	1	1066
2	1	1	-	1066
		-	1	1066
	2	2	-	1066
		1	1	1066
		-	2	1066

1. Population restrictions (including the order for partially populated channels) may apply. See Table 12.

Table 11: DDR3 SO-DIMM Maximum Frequency Support for FT1

DIMM Slots/Ch	DIMMs	DIMMs		Frequency[1] (MT/s)
		SR	DR	1.5V / 1.35 V
1	1	1	-	1066
		-	1	1066
2	1	1	-	1066
		-	1	1066
	2	2	-	1066
		1	1	1066
		-	2	1066

1. Population restrictions (including the order for partially populated channels) may apply. See Table 12.

The tables below list the DIMM populations as supported by the processor. DIMMs must be populated from

farthest slot to closest slot to the processor when a daisy chain topology is used.

Table 12: DDR3 UDIMM and SO-DIMM Population Support for FT1

DIMM Slots/Ch	DIMM0	DIMM1
1	SR/DR	N/A
2	-	SR/DR
	SR/DR	SR/DR

Figure 6 shows a channel with two dual rank DIMMs.

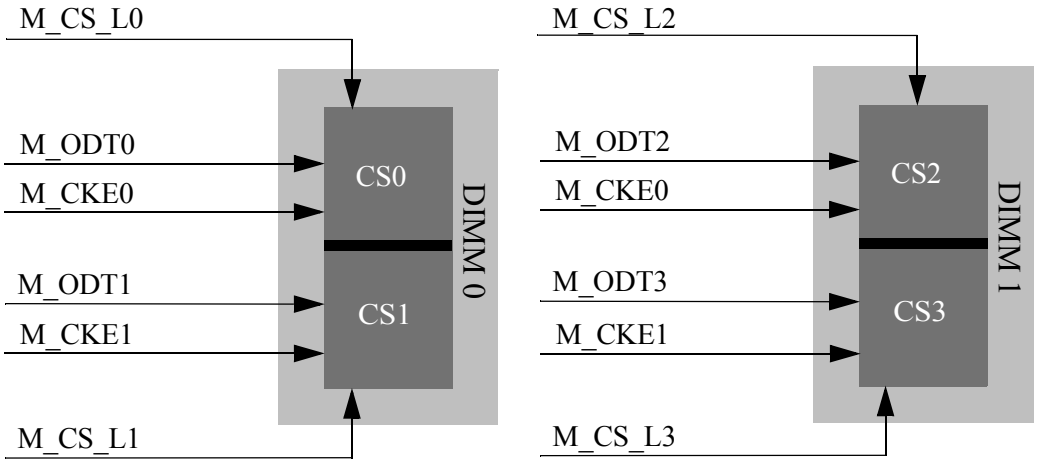


Figure 6: Control Pin Configuration with 2 dual rank DIMMs

2.9.1 DCT Configuration Registers

The DCT configuration registers reside in device 18h function 2 configuration space.

A subset of DCT configuration registers must be programmed for each supported NB P-state. See 2.9.3.4.7 [NB P-states for DCT/DRAM Initialization and Training].

2.9.2 DRAM Controller Direct Response Mode

The DCT supports direct response mode for responding to a cache line fill request before the DCT is initialized. In direct response mode, the target DCT responds to a cache line fill request by returning 64 bytes of all ones without issuing a read transaction on the DRAM bus. The BIOS uses this feature to allocate cache lines for temporary data storage. The controller exits direct response mode when D18F2x7C[EnDramInit] is set to 1.

See 2.9.3.6 [DRAM Device Initialization] and 2.3.3 [Cache Initialization For General Storage During Boot].

2.9.3 DCT/DRAM Initialization and Resume

DRAM initialization requires several steps to configure the DCT and DIMMs, as well as tuning the DRAM channel to ensure stability. DRAM resume requires several steps to configure the DCT to properly resume from the S3 state. The following sequence describes the steps needed to enable the DRAM channel after a reset for initialization or resume:

1. Configure the DDR supply voltage regulator. See 2.9.3.1.
2. Force NB P-state to NBP0. See 2.5.4.1.3.
   A. Program D18F6x90[NbPsCtrlDis]=1.
   B. Program D18F6x90[NbPsForceSel]=0.
   C. Program D18F6x90[NbPsForceReq]=1.
   D. Wait for D18F6x98[NbPs1Act]=0.
3. DDR phy initialization. See 2.9.3.1.
4. DRAM device and controller initialization.
   - If BIOS is booting from an unpowered state (ACPI S4, S5 or G3), then it performs the following:
     a. Program SPD configuration. See 2.9.3.3.
     b. Program Non-SPD configuration. See 2.9.3.4.
     c. Program NBP0 specific configuration. See 2.9.3.4.7.
     d. DRAM device initialization. See 2.9.3.6.
     e. Program DCT training specific configuration. See 2.9.3.5.
   - If BIOS is resuming the platform from S3 state, then it performs the following:
     a. Restore all DCT and phy registers that were programmed during the first boot from non-volatile storage, including NBP0 and NBP1 version of registers specified in 2.9.3.4.7. Use D18F6x98[NbPsDbgEn, NbPsCsrAccSel] to restore NBP0/NBP1 specific registers. See 2.9.3.3, 2.9.3.4, and 2.9.3.4.7 for a review of registers.
     b. Program D18F2x90[ExitSelfRef] = 1.
     c. Restore the trained delayed values (found during the initial boot in steps 4 and 5 below) from non-volatile storage.
     d. Continue at step 10.
5. DRAM write levelization training. See 2.9.3.7.1.
6. DRAM data training.
   A. DQS receiver enable training. See 2.9.3.7.2.
   B. Program D18F2x9C_x0D0F_E003[DisAutoComp, DisablePredriverCal] = {0b, 1b}.
   C. Program D18F2x78[RxPtrInitReq]=1.
   D. Program D18F2xA8[DbeGskMemClkAlignMode] as follows:
      a. Program D18F2x90[DisDllShutDownSR] = 1.
      b. Program D18F2x90[EnterSelfRef] = 1.
      c. Wait for D18F2x90[EnterSelfRef] = 0.
      d. Program D18F2xA8[DbeGskMemClkAlignMode] = 10b.
      e. Program D18F2x90[ExitSelfRef] = 1.
      f. Wait for D18F2x90[ExitSelfRef] = 0.
      g. Program D18F2x90[DisDllShutDownSR] = 0.
   E. DQS position training. See 2.9.3.7.3.
   F. MaxRdLatency training. See 2.9.3.7.4.1.

IF (D18F6x90[NbPsCap]=1) THEN

7. Program NBP1 specific configuration. See 2.9.3.4.7. Use D18F6x98[NbPsDbgEn, NbPsCsrAccSel] to select NBP1 registers.
8. Force NB P-state to NBP1. See 2.5.4.1.3.
   A. Program D18F6x90[NbPsForceSel]=1.
   B. Wait for D18F6x98[NbPs1Act]=1.

9. Perform MaxRdLatency training for NBP1. See 2.9.3.7.4.1.

ENDIF.

10. Release NB P-state force. See 2.5.4.1.3.
    A. Program D18F6x90[NbPsForceReq]=0.
    B. Program D18F6x90[NbPsCtrlDis]=0.
11. Program DCT for normal operation. See 2.9.3.5.
12. Program DRAM Phy for power savings. See 2.9.3.8.

The DRAM subsystem is ready for use.

2.9.3.1 Low Voltage DDR3

JEDEC defined 1.5 V and 1.35 V DDR3 devices are supported. Platforms that support 1.35 V operation should power on VDDIO_MEM_S at 1.35 V until operating voltage is determined by reading the SPD ROM of all the DIMMs. BIOS should not operate DIMMs at voltages higher than supported as indicated by the SPD.

The recommended BIOS configuration sequence is as follows:

1. BIOS reads the SPD ROM of all DIMMs to determine the common operating voltages.
2. BIOS configures VDDIO_MEM_S to match the lowest common supported voltage based on the SPD values. See platform specific documentation for changing the voltage.

2.9.3.2 DDR Phy Initialization

The BIOS initializes the phy and the internal interface from the DCT to the phy, including the PLLs and the fence value, after each reset and for each time a frequency change is made.

BIOS obtains size, loading, and frequency information about the DIMMs and channels using SPDs prior to phy initialization. BIOS then performs the following actions:

1. Program D18F2x9C_x0000_000B = 80000000h.
2. Phy Voltage Level Programming. See 2.9.3.2.1.
3. DRAM channel frequency change. See 2.9.3.2.2.
4. If BIOS is booting from an unpowered state (ACPI S4, S5 or G3; not S3, suspend to RAM), then it performs the following:
   A. Program D18F2xA8[DbeGskMemClkAlignMode] = 00b.
   B. Program D18F2x7C[EnDramInit] = 1.
5. Phy fence programming. See 2.9.3.2.3.
6. Phy compensation initialization. See 2.9.3.2.4.

2.9.3.2.1 Phy Voltage Level Programming

BIOS programs the following according to the desired phy VDDIO_MEM_S voltage level:

- Program D18F2x9C_x0D0F_0[F,7:0]1F[RxVioLvl].
- Program D18F2x9C_x0D0F_[C,8,2][1:0]1F[RxVioLvl].
- Program D18F2x9C_x0D0F_4009[CmpVioLvl].

See 2.9.3.1 [Low Voltage DDR3].

2.9.3.2.2 DRAM Channel Frequency Change

The following sequence is used to change the DRAM frequency under all boot conditions, including restoring the DCT state when resuming from the S3 state:

1. Program D18F2x9C_x0D0F_E006[PllLockTime] = 1838h.
2. Program D18F2x94[MemClkFreqVal] = 0.
3. Program D18F2x94[MemClkFreq] to the desired DRAM frequency.
4. Configure NBP0 and NBP1 frequency and voltage to meet NCLK-MEMCLK ratio requirements for the desired DRAM frequency. See 2.5.4.1.1 [BIOS Requirements for NB P-state Initialization During DRAM Training].
5. Program the following according to the new MemClkFreq value:
   A. Program D18F2xF4_x30[DbeGskFifoNumerator] = NclkFid * MemClkDid * 16.
      - NclkFid = NCLK PLL multiplier as defined by D18F3xD4[MainPllOpFreqId] (i.e. COF/100 MHz).
      - MemClkDid = MEMCLK PLL divide ratio as defined by Table 58.

   B. Program D18F2xF4_x31[DbeGskFifoDenominator] = NclkDiv * PllMult.
      - NclkDiv = If programming NBP1 then D18F6x90[NbPs1NclkDiv], else D18F3xDC[NbPs0NclkDiv].
      - PllMult = MEMCLK PLL multiplier as defined by Table 58.

   C. Program D18F2xF4_x32[DataTxFifoSchedDlyNegSlot1, DataTxFifoSchedDlySlot1, DataTxFifoSchedDlyNegSlot0, DataTxFifoSchedDlySlot0]. See 2.9.3.2.2.2 [DCT Transmit FIFO Schedule Delay Programming].
   D. IF (D18F2x94[MemClkFreq] >= 667 MHz) THEN
      Program D18F2x78[RdPtrInit] = 2
      ELSE
      Program D18F2x78[RdPtrInit] = 3
      ENDIF.
   E. Program D18F2x9C_x0D0F_0[F,7:0]13[ProcOdtAdv] = 1.
6. Program D18F2x94[MemClkFreqVal] = 1.
7. IF (D18F2x9C_x0D0F_E00A[CsrPhySrPllPdMode]==0) THEN Program D18F2x9C_x0D0F_E006[PllLockTime] = 0Fh. ENDIF.

2.9.3.2.2.1 Requirements for DRAM Frequency Change During Training

During DRAM training, BIOS may be required to change the DRAM(MEMCLK) frequency. The steps below describe what is required to prepare the processor and memory subsystem for the new MEMCLK frequency. It is assumed that the memory subsystem has previously been initialized at the current MEMCLK frequency, and this procedure describes only the steps that must be repeated at the new MEMCLK frequency. See 2.9.3.7.1 [Write Levelization Training] and 2.9.3.7.2 [DQS Receiver Enable Training].

1. Ensure NB P-states are disabled prior to this procedure. See D18F6x90[NbPsCtrlDis].
2. Enter self-refresh:
   A. Program D18F2x90[DisDllShutDownSR] = 1.
   B. Program D18F2x90[EnterSelfRef] = 1.
   C. Wait for D18F2x90[EnterSelfRef] = 0.
3. DRAM channel frequency change. See 2.9.3.2.2.
4. Exit self-refresh:
   A. Program D18F2x90[ExitSelfRef] = 1.
   B. Wait for D18F2x90[ExitSelfRef] = 0.
   C. Program D18F2x90[DisDllShutDownSR] = 0.
5. Phy fence programming. See 2.9.3.2.3.

6. Phy compensation initialization. See 2.9.3.3.
7. Program SPD configuration. See 2.9.3.3.
8. Program Non-SPD configuration. See 2.9.3.4.
9. Program NB P-state specific configuration. See 2.9.3.4.7.
10. Issue MRS(2), MRS(3), MRS(1), MRS(0) commands. See 2.9.3.6.1 [Software DDR3 Device Initialization].

2.9.3.2.2.2 DCT Transmit FIFO Schedule Delay Programming

The optimal value for D18F2xF4_x32[DataTxFifoSchedDlyNegSlot1, DataTxFifoSchedDlySlot1, DataTxFifoSchedDlyNegSlot0, DataTxFifoSchedDlySlot0] is configuration specific. BIOS should use the guidelines below to configure the recommended values:

For N=0,1:
- If (PartialSumSlotN >= 0):
  - DataTxFifoSchedDlySlotN=CEIL(PartialSumSlotN).
  - DataTxFifoSchedDlyNegSlotN=0.
- Else if (PartialSumSlotN < 0):
  - DataTxFifoSchedDlySlotN=ABS(CEIL(PartialSumSlotN*MemClkPeriod/NclkPeriod)).
  - DataTxFifoSchedDlyNegSlotN=1.

- PartialSumSlot0 = (((5 * NclkPeriod[1]) + MemClkPeriod[2] + 520 ps) * MemClkFrequency) - tCWL[3] - CmdSetup[4] - SlowAccessMode[5] - PtrSeparation[6].
- PartialSumSlot1 = (((5 * NclkPeriod[1]) + MemClkPeriod[2] + 520 ps) * MemClkFrequency) - tCWL[3] - CmdSetup[4] - 1 - PtrSeparation[6].

Notes:
1. NclkPeriod = NCLK period as defined by D18F3xDC[NbPs0NclkDiv] or D18F6x90[NbPs1NclkDiv] for given NB P-state.
2. MemClkPeriod/MemClkFrequency = MEMCLK period/frequency as defined by D18F2x94[MemClkFreq].
3. tCWL = Tcwl in MEMCLKs as defined by D18F2x84[Tcwl].
4. CmdSetup = 1/2 MEMCLK if all of D18F2x9C_x0000_0004[AddrCmdSetup, CsOdtSetup, CkeSetup]=0, else 1 MEMCLK.
5. SlowAccessMode = 1 MEMCLK if D18F2x94[SlowAccessMode]=1, else 0.
6. PtrSeparation:
   - PtrSeparation = ((16 + RdPtrInitMin - D18F2x78[RdPtrInit]) MOD 16)/2 + RdPtrInitRmdr.
   - If (D18F2x94[MemClkFreq] >= 667 MHz)
     then RdPtrInitMin = 2
     else RdPtrInitMin = 3.
   - RdPtrInitRmdr = (((2.25 * MemClkPeriod) - 1520ps) MOD MemClkPeriod)/MemClkPeriod.

2.9.3.2.3 Phy Fence Programming

The DDR phy fence logic is used to adjust the phase relationship between the data FIFO and the data going to the pad. After any MEMCLK frequency change and before any memory training, BIOS must perform phy fence training using the following steps:

1. Program D18F2x9C_x0000_0008[FenceTrSel]=10b.
2. Program D18F2x9C_x0000_00[51:50]=1313_1313h.
3. Perform phy fence training. See 2.9.3.2.3.1 [Phy Fence Training].

4. Write the calculated fence value to D18F2x9C_x0000_000C[FenceThresholdTxDll].
5. Program D18F2x9C_x0D0F_0[F,7:0]0F[AlwaysEnDllClks]=001b.
6. Program D18F2x9C_x0000_0008[FenceTrSel]=01b.
7. Program D18F2x9C_x0000_00[51:50]=1313_1313h.
8. Perform phy fence training. See 2.9.3.2.3.1 [Phy Fence Training].
9. Write the calculated fence value to D18F2x9C_x0000_000C[FenceThresholdRxDll].
10. Program D18F2x9C_x0D0F_0[F,7:0]0F[AlwaysEnDllClks]=000b.
11. Program D18F2x9C_x0000_0008[FenceTrSel]=11b.
12. Program D18F2x9C_x0000_00[51:50]=1313_1313h.
13. Perform phy fence training. See 2.9.3.2.3.1 [Phy Fence Training].
14. Write the calculated fence value to D18F2x9C_x0000_000C[FenceThresholdTxPad].
15. IF (D18F2x9C_x0000_000C[FenceThresholdTxPad] < 16) THEN
    Program D18F2x9C_x0D0F_[C,8,2][1:0]31 = {001h, D18F2x9C_x0000_000C[19:16]}
    ELSE
    Program D18F2x9C_x0D0F_[C,8,2][1:0]31 = 0000h
    ENDIF.
16. Program Fence2 threshold for data as follows:
    A. IF (D18F2x9C_x0000_000C[FenceThresholdTxPad] < 16) THEN
       Fence2_TxPad[4:0] = {1b, D18F2x9C_x0000_000C[19:16]}
       ELSE
       Fence2_TxPad[4:0] = 00000b
       ENDIF.
    B. IF (D18F2x9C_x0000_000C[FenceThresholdRxDll] < 16) THEN
       Fence2_RxDll[4:0] = {1b, D18F2x9C_x0000_000C[24:21]}
       ELSE
       Fence2_RxDll[4:0] = 00000b
       ENDIF.
    C. IF (D18F2x9C_x0000_000C[FenceThresholdTxDll] < 16) THEN
       Fence2_TxDll[4:0] = {1b, D18F2x9C_x0000_000C[29:26]}
       ELSE
       Fence2_TxDll[4:0] = 00000b
       ENDIF.
    D. Program D18F2x9C_x0D0F_0[F,7:0]31 = {0b, Fence2_RxDll[4:0], Fence2_TxDll[4:0], Fence2_TxPad[4:0]}.
17. Reprogram D18F2x9C_x0000_0004.

When resuming from S3, it is recommended that BIOS reprogram D18F2x9C_x0000_000C[FenceThresholdTxDll, FenceThresholdRxDll, FenceThresholdTxPad], D18F2x9C_x0D0F_[C,8,2][1:0]31, and D18F2x9C_x0D0F_0[F,7:0]31 from values stored in non-volatile storage instead of training.

2.9.3.2.3.1 Phy Fence Training

The following describes the steps for each pass of phy fence training:

1. Program D18F2x9C_x0000_0008[PhyFenceTrEn]=1.
2. Wait 2000 MEMCLKs.
3. Program D18F2x9C_x0000_0008[PhyFenceTrEn]=0.
4. Read the phase recovery engine registers D18F2x9C_x0000_00[51:50].
5. Calculate the fence value by averaging the fine delay values of all byte lanes and subtract 8.

2.9.3.2.4 Phy Compensation Initialization

Each DDR IO driver has a programmable slew rate controlled by the pre-driver calibration code. The recommended slew rate is a function of the DC drive strength. BIOS initializes the recommended nominal slew rate values as follows:

1. Program D18F2x9C_x0D0F_E003[DisAutoComp, DisablePreDriverCal] = {1b, 1b}.
2. Program TxPreP/TxPreN for Data and DQS according to Table 13 if VDDIO_MEM_S is 1.5V or Table 14 if 1.35V.
   A. Program D18F2x9C_x0D0F_0[F,7:0]0[A,6]={0000b, TxPreP, TxPreN}.
   B. Program D18F2x9C_x0D0F_0[F,7:0]02={1000b, TxPreP, TxPreN}.
3. Program TxPreP/TxPreN for Cmd/Addr according to Table 15 if VDDIO_MEM_S is 1.5V or Table 16 if 1.35V.
   A. Program D18F2x9C_x0D0F_[C,8][1:0][12,0E,0A,06]={0000b, TxPreP, TxPreN}.
   B. Program D18F2x9C_x0D0F_[C,8][1:0]02={1000b, TxPreP, TxPreN}.
4. Program TxPreP/TxPreN for Clock according to Table 17 if VDDIO_MEM_S is 1.5V or Table 18 if 1.35V.
   A. Program D18F2x9C_x0D0F_2[1:0]02={1000b, TxPreP, TxPreN}.

Table 13: Phy pre-driver calibration codes for Data/DQS at 1.5V

DDR Rate	Drive Strength[1]	TxPreP[2]	TxPreN[2]
800	000b	100_100b	100_100b
	001b	100_100b	100_100b
	010b	100_100b	100_100b
	011b	100_100b	100_100b
1066	000b	111_111b	110_110b
	001b	111_111b	110_110b
	010b	111_111b	110_110b
	011b	111_111b	110_110b
1. IF (D18F2x9C_x0D0F_0[F,7:0]06) THEN See D18F2x9C_x0000_0000[DqsDrvStren] ELSE See D18F2x9C_x0000_0000[DataDrvStren] ENDIF. 2. See D18F2x9C_x0D0F_0[F,7:0]0[A,6] and D18F2x9C_x0D0F_0[F,7:0]02.			

Table 14: Phy pre-driver calibration codes for Data/DQS at 1.35V

DDR Rate	Drive Strength[1]	TxPreP[2]	TxPreN[2]
800	000b	111_111b	110_110b
	001b	101_101b	101_101b
	010b	101_101b	101_101b
	011b	100_100b	100_100b

Table 14: Phy pre-driver calibration codes for Data/DQS at 1.35V

DDR Rate	Drive Strength[1]	TxPreP[2]	TxPreN[2]
1066	000b	111_111b	110_110b
	001b	111_111b	110_110b
	010b	111_111b	110_110b
	011b	111_111b	110_110b

1. IF (D18F2x9C_x0D0F_0[F,7:0]06) THEN
   See D18F2x9C_x0000_0000[DqsDrvStren]
   ELSE
   See D18F2x9C_x0000_0000[DataDrvStren]
   ENDIF.
2. See D18F2x9C_x0D0F_0[F,7:0]0[A,6] and D18F2x9C_x0D0F_0[F,7:0]02.

Table 15: Phy pre-driver calibration codes for Cmd/Addr at 1.5V

DDR Rate	Drive Strength[1]	TxPreP[2]	TxPreN[2]
800	000b	010_010b	010_010b
	001b	010_010b	010_010b
	010b	010_010b	010_010b
	011b	010_010b	010_010b
1066	000b	011_011b	011_011b
	001b	011_011b	011_011b
	010b	011_011b	011_011b
	011b	011_011b	011_011b

1. IF (D18F2x9C_x0D0F_800[A,6,2])THEN
   See D18F2x9C_x0000_0000[CsOdtDrvStren]
   ELSE
   See D18F2x9C_x0000_0000[AddrCmdDrvStren]
   ENDIF.
2. See D18F2x9C_x0D0F_[C,8][1:0][12,0E,0A,06] and D18F2x9C_x0D0F_[C,8][1:0]02.

Table 16: Phy pre-driver calibration codes for Cmd/Addr at 1.35V

DDR Rate	Drive Strength[1]	TxPreP[2]	TxPreN[2]
800	000b	010_010b	010_010b
	001b	010_010b	010_010b
	010b	010_010b	010_010b
	011b	010_010b	010_010b

Table 16: Phy pre-driver calibration codes for Cmd/Addr at 1.35V

DDR Rate	Drive Strength[1]	TxPreP[2]	TxPreN[2]
1066	000b	100_100b	100_100b
	001b	011_011b	011_011b
	010b	011_011b	011_011b
	011b	011_011b	011_011b

1. IF (D18F2x9C_x0D0F_800[A,6,2])THEN
   See D18F2x9C_x0000_0000[CsOdtDrvStren]
   ELSE
   See D18F2x9C_x0000_0000[AddrCmdDrvStren]
   ENDIF.
2. See D18F2x9C_x0D0F_[C,8][1:0][12,0E,0A,06] and D18F2x9C_x0D0F_[C,8][1:0]02.

Table 17: Phy pre-driver calibration codes for Clock at 1.5V

DDR Rate	Drive Strength[1]	TxPreP[2]	TxPreN[2]
800	000b	100_100b	100_100b
	001b	100_100b	100_100b
	010b	100_100b	100_100b
	011b	100_100b	100_100b
1066	000b	111_111b	110_110b
	001b	111_111b	110_110b
	010b	111_111b	110_110b
	011b	101_101b	101_101b

1. See D18F2x9C_x0000_0000[ClkDrvStren].
2. See D18F2x9C_x0D0F_2[1:0]02.

Table 18: Phy pre-driver calibration codes for Clock at 1.35V

DDR Rate	Drive Strength[1]	TxPreP[2]	TxPreN[2]
800	000b	110_110b	101_101b
	001b	110_110b	101_101b
	010b	100_100b	100_100b
	011b	100_100b	100_100b
1066	000b	111_111b	110_110b
	001b	111_111b	110_110b
	010b	111_111b	110_110b
	011b	110_110b	101_101b

1. See D18F2x9C_x0000_0000[ClkDrvStren].
2. See D18F2x9C_x0D0F_2[1:0]02.

2.9.3.3 SPD ROM-Based Configuration

The Serial Presence Detect (SPD) ROM is a non-volatile memory device on the DIMM encoded by the DIMM

manufacturer. The description of the SPD is usually provided on a data sheet for the DIMM itself along with data describing the memory devices used. The data describes configuration and speed characteristics of the DIMM and the SDRAM components mounted on the DIMM. The associated data sheet also contains the DIMM byte values that are encoded in the SPD on the DIMM.

BIOS reads the values encoded in the SPD ROM through a system-specific interface. BIOS acquires DIMM configuration information, such as the amount of memory on each DIMM, from the SPD ROM on each DIMM and uses this information to program the DRAM controller registers.

The SPD ROM provides values for several DRAM timing parameters that are required by the DCT. In general, BIOS should use the optimal value specified by the SPD ROM. These parameters are:

- D18F2x84[Twr]: Write recovery time
- D18F2x88[Tcl]: CAS latency
- D18F2x8C[Tref]: Refresh interval
- D18F2x8C[Trfc1, Trfc0]: Auto Refresh to Active/Auto Refresh delay
- D18F2x94[FourActWindow]: Four activate window delay time
- D18F2xF4_x40[Tras]: Active to Precharge delay
- D18F2xF4_x40[Trc]: Active to Active/Auto Refresh delay
- D18F2xF4_x40[Trcd]: RAS to CAS delay
- D18F2xF4_x40[Trp]: Precharge time
- D18F2xF4_x41[Trrd]: Active-Bank-A to Active-Bank-B delay
- D18F2xF4_x41[Trtp]: Internal Read to Precharge command delay
- D18F2xF4_x41[Twtr]: Internal write to read command delay

Optimal cycle time is specified for each DIMM and is used to limit or determine bus frequency. See 2.9.3.6 [DRAM Device Initialization].

2.9.3.3.1 FourActWindow (Four Bank Activate Window or tFAW)

No more than 4 banks may be activated in a rolling tFAW window, as configured by D18F2x94[FourActWindow]. To program this field, BIOS must convert the tFAW parameter into MEMCLK cycles by dividing the highest tFAW parameter (in ns) found in all the DIMMs connected to the channel by the period of MEMCLK (in ns) and rounding up to the next integer.

2.9.3.4 Non-SPD ROM-Based Configuration

There are several DRAM timing parameters and DCT configurations that need to be programmed for optimal memory performance. These values are not derived from the SPD ROM. Several of these timing parameters are functions of other configuration values. These interdependencies must be considered when programming values into several DCT register timing fields. The factors to consider when specifying a value for a specific non-SPD timing parameter are:

- Training delay values. See 2.9.3.7 [DRAM Training].
- Read and write latency differences.
- The phy's idle clock requirements on the data bus.
- DDR3 ODT timing requirements.
- NCLK frequency for each supported NB P-state.
- MEMCLK frequency.

The following sub-sections describe how BIOS programs each non-SPD related timing field to a recommended

minimum timing value with respect to the above factors.

The following terms are defined to simplify calculations and are calculated in MEMCLKs:
- Latency Difference (LD) = D18F2x88[Tcl] - D18F2x84[Tcwl].
- Read ODT Delay (ROD) = MAX(0, D18F2xF4_x83[RdOdtOnDuration] - 6).
- Write ODT Delay (WOD) = MAX(0, D18F2xF4_x83[WrOdtOnDuration] - 6).

2.9.3.4.1 Trdrd and TrdrdSD (Read-to-Read Timing)

The optimal values for D18F2x8C[Trdrd] and D18F2xF4_x06[TrdrdSD] are platform and configuration specific and should be characterized for best performance. Prior to DRAM training, BIOS should program these parameters to the largest defined value. After DRAM training, BIOS should use the guidelines below to configure the recommended platform generic timing values:

- TrdrdSD (in MEMCLKs) = 3.
- Trdrd (in MEMCLKs) = CEIL(MAX(ROD + 3, CDDTrdrd/2 + (D18F2x[94]SlowAccessMode ? 3 : 3.5))).

The Critical Delay Difference (CDD) is the largest delay difference of the channel.
- Each delay difference is D18F2x9C_x0000_00[24:10][DqsRcvEnGrossDelay] minus D18F2x9C_x0000_00[24:10][DqsRcvEnGrossDelay].
- For CDD_{Trdrd}, the subtraction terms are the delays of different DIMMs within the same byte lane.

BIOS must program these parameters as follows: TrdrdSD <= Trdrd.

2.9.3.4.2 Twrwr and TwrwrSD (Write-to-Write Timing)

The optimal values for D18F2x8C[Twrwr] and D18F2xF4_x16[TwrwrSD] are platform and configuration specific and should be characterized for best performance. Prior to DRAM training, BIOS should program these parameters to the largest defined value. After DRAM training, BIOS should use the guidelines below to configure the recommended platform generic timing values:

- TwrwrSD (in MEMCLKs) = WOD + 3.
- Twrwr (in MEMCLKs) = CEIL(MAX(WOD + 3, CDD_{Twrwr} / 2 + 3.5)).

The Critical Delay Difference (CDD) is the largest delay difference of the channel.
- Each delay difference is D18F2x9C_x0000_00[44:30][WrDqsGrossDly] minus D18F2x9C_x0000_00[44:30][WrDqsGrossDly].
- For CDD_{Twrwr}, the subtraction terms are the delays of different DIMMs within the same byte lane.

BIOS must program these parameters as follows: TwrwrSD <= Twrwr.

2.9.3.4.3 Twrrd and TwrrdSD (Write-to-Read DIMM Termination Turn-around)

The optimal value for D18F2x8C[Twrrd] and D18F2xF4_x06[TwrrdSD] is platform and configuration specific and should be characterized for best performance. Prior to DRAM training, BIOS should program these parameters to the largest defined value. After DRAM training, BIOS should use the guidelines below to configure the recommended platform generic timing values:

- TwrrdSD (in MEMCLKs) = CEIL(MAX(1, MAX(WOD, $CDD_{TwrrdSD}$ / 2 + 0.5) - LD + 3)).
- Twrrd (in MEMCLKs) = CEIL(MAX(1, MAX(WOD, CDD_{Twrrd} / 2 + 0.5) - LD + 3)).

The Critical Delay Difference (CDD) is the largest delay difference of the channel.
- Each delay difference is D18F2x9C_x0000_00[44:30][WrDqsGrossDly] minus D18F2x9C_x0000_00[24:10][DqsRcvEnGrossDelay].
- For $CDD_{TwrrdSD}$, the subtraction terms are the delays of the same DIMM within the same byte lane.
- For CDD_{Twrrd}, the subtraction terms are the delays of different DIMMs within the same byte lane.

BIOS must program these parameters as follows: TwrrdSD <= Twrrd.

2.9.3.4.4 TrwtTO (Read-to-Write Turnaround for Data, DQS Contention)

The optimal value for D18F2x8C[TrwtTO] is platform and configuration specific and should be characterized for best performance. Prior to DRAM training, BIOS should program this parameter to the largest defined value. After DRAM training, BIOS should use the guidelines below to configure the recommended platform generic timing values after DDR training is complete:

- TrwtTO (in MEMCLKs) = CEIL(MAX(ROD, CDD_{TrwtTO} / 2 - 0.5) + LD + 3).
  - If 1 DIMM per channel, substitute ROD=0.

The Critical Delay Difference (CDD) is the largest delay difference of the channel.
- Each delay difference is D18F2x9C_x0000_00[24:10][DqsRcvEnGrossDelay] minus D18F2x9C_x0000_00[44:30][WrDqsGrossDly].
- For CDD_{TrwtTO}, the subtraction terms are the delays of all DIMMs within the same byte lane.

2.9.3.4.5 DRAM ODT Control

This section describes the ODT configurations and settings for the processor and attached DIMMs. The tables specify ODT values for different configurations. The DIMM termination values are programmed as specified below during DDR3 device initialization. If the DIMM termination values are changed after device initialization then BIOS must issue MRS commands to the devices to change the values. See 2.9.3.6.1 [Software DDR3 Device Initialization].

Table 19 specifies the ODT nominal (non-write) and dynamic termination resistance values for different DIMM configurations.

BIOS configures the ODT turn on delay and duration for reads and writes. See D18F2xF4_x83[RdOdtTrnOnDly, RdOdtOnDuration, WrOdtTrnOnDly, WrOdtOnDuration].

Table 19: DIMM ODT settings

DDR Rate	DIMMs Populated[1]	DIMM ODT (Rtt_Nom)	DIMM Dynamic ODT (Rtt_Wr)
800	1	120 ohms	Disabled
1066	1	120 ohms	Disabled
800	2	40 ohms	120 ohms
1066	2	IF SO-DIMM 30 ohms, ELSE 40 ohms	120 ohms

1. DIMMs can be single or dual rank. DIMMs can be unbuffered or SO- DIMMs.

The following describes the general ODT behavior for various system configurations. In all cases, the proces-

sor ODT is off for writes and is on for reads:

- For one single or dual rank DIMM on the channel:
  - For writes, the ODT is on for the target rank.
  - For reads, the ODT is off for all ranks.
- For two single or dual rank DIMMs on the channel:
  - For writes, the ODT is on for the target rank of the target DIMM and also on for the first rank of the non-target DIMM.
  - For reads, the ODT is on for the first rank of the non-target DIMM.

BIOS configures the DIMM ODT behavior on a per chip select basis according to the DIMM population. The ODT patterns for reads and writes are programmed using D18F2xF4_x180 and D18F2xF4_x182, respectively, as specified by Table 20. BIOS also configures the DIMM ODT pattern used during write levelization training by setting D18F2x9C_x0000_0008[WrLvOdtEn] and programming D18F2x9C_x0000_0008[WrLvOdt]. BIOS programs D18F2x9C_x0000_0008[WrLvOdt] with the D18F2xF4_x182 value provided for writes to the rank targeted by training. See 2.9.3.7.1 [Write Levelization Training].

Table 20: DIMM ODT pattern

DIMM0[1]	DIMM1[1]	D18F2xF4_x180	D18F2xF4_x182
SR[2]	-	0000_0000h	0000_0001h
DR[2]	-	0000_0000h	0000_0201h
-	SR	0000_0000h	0004_0000h
-	DR	0000_0000h	0804_0000h
SR/DR	SR/DR	0101_0404h	0905_0605h

1. SR = Single rank, DR = Dual rank.
2. Only supported in systems which support a single DIMM on the channel.

2.9.3.4.6 DRAM Address Timing and Output Driver Compensation Control

This section describes the settings required for programming the timing on the address pins, the CS/ODT pins, and the CKE pins, as well as the processor ODT values controlled by D18F2x9C_x0000_0000[ProcOdt] and D18F2x9C_x0D0F_0[F,7:0]0[8,0][ProcOdtDqOvrd]. Table 21 and Table 22 document the address timing, output driver settings, and processor ODT values for different DDR DIMM types. The DIMMs are numbered from 0 to n where DIMM0 is the DIMM closest to the processor on the channel and DIMMn is the DIMM farthest from the processor on the channel. DIMMs must be populated from farthest slot to closest slot to the processor. Populations that are not shown in these tables are not supported. These tables document the optimal settings for motherboards which meet the relevant motherboard design guidelines.

Table 21: BIOS Recommendations **for SO-DIMM address timings and output driver control**

Condition					D18F2x94	D18F2x9C_x0000_0004	D18F2x9C_x0000_0000	D18F2x9C_x0D0F_0[F,7:0][0[8,0][ProcOdtDqOvrd]
NumDimmSlots	DdrRate	VDDIO_MEM_S	DIMM0	DIMM1	SlowAccessMode			
1	800	1.35, 1.5	SR	-	0	00000000h	00002223h	0h
1	800	1.35, 1.5	DR	-	0	00000000h	00002223h	0h
1	1066	1.35, 1.5	SR	-	0	003D3D3Dh	10002223h	1h
1	1066	1.35, 1.5	DR	-	0	00000000h	10002223h	1h
2	800	1.35, 1.5	NP	SR	0	00000000h	00002223h	0h
2	800	1.35, 1.5	NP	DR	0	00000000h	00002223h	0h
2	800	1.35, 1.5	SR, DR	SR, DR	1	00000039h	20222323h	2h
2	1066	1.35, 1.5	NP	SR	0	003D3D3Dh	10002223h	1h
2	1066	1.35, 1.5	NP	DR	0	00000000h	10002223h	1h
2	1066	1.35, 1.5	SR, DR	SR, DR	1	00000037h	30222323h	3h

Table 22: BIOS Recommendations **for UDIMM address timings and output driver control**

Condition					D18F2x94	D18F2x9C_x0000_0004	D18F2x9C_x0000_0000	D18F2x9C_x0D0F_0[F,7:0][0[8,0][ProcOdtDqOvrd]
NumDimmSlots	DdrRate	VDDIO_MEM_S	DIMM0	DIMM1	SlowAccessMode			
1	800	1.5	SR	-	0	00000000h	00112223h	0h
1	800	1.5	DR	-	0	003B0000h	00112223h	0h
1	1066	1.5	SR	-	0	00000000h	10112223h	1h
1	1066	1.5	DR	-	0	00380000h	10112223h	1h
2	800	1.5	NP	SR	0	00000000h	00112223h	0h
2	800	1.5	NP	DR	0	003B0000h	00112223h	0h
2	800	1.5	SR, DR	SR, DR	0	00390039h	20222323h	2h
2	1066	1.5	NP	SR	0	00000000h	10112223h	1h
2	1066	1.5	NP	DR	0	00380000h	10112223h	1h
2	1066	1.5	SR, DR	SR, DR	0	00350037h	30222323h	3h

2.9.3.4.7 NB P-states for DCT/DRAM Initialization and Training

Before DRAM device initialization and training or prior to register restore when resuming the platform from S3, BIOS must force the processor to the NBP0 P-state. A subset of initialization and training must be repeated while forcing the processor to the NBP1 P-state. When DRAM training is complete, BIOS releases the force on the NB P-state. See 2.5.4.1.3 [Software Controlled NB P-states] and 2.9.3 [DCT/DRAM Initialization and Resume].

The following configuration registers contain multiple internal copies and must be programmed multiple times, once for each supported NB P-state. See D18F6x98[NbPsDbgEn, NbPsCsrAccSel] for information regarding how register context is selected.

- D18F2x78[RdPtrInit].
- D18F2x78[DisCutThroughMode].
- D18F2x78[ForceCasToSlot0].
- D18F2x78[MaxRdLatency]. See 2.9.3.7.4.1 [MaxRdLatency Training].
- D18F2xF4_x30[DbeGskFifoNumerator].

- D18F2xF4_x31[DbeGskFifoDenominator].
- D18F2xF4_x32[DataTxFifoSchedDlySlot1].
- D18F2xF4_x32[DataTxFifoSchedDlyNegSlot1].
- D18F2xF4_x32[DataTxFifoSchedDlySlot0].
- D18F2xF4_x32[DataTxFifoSchedDlyNegSlot0].

2.9.3.5 DCT Training Specific Configuration

The DCT requires certain features be disabled during DRAM device initialization and training. BIOS should program the registers in Table 23 before and after DRAM device initialization and training. BIOS must quiesce all other forms of DRAM traffic on the channel being trained. See 2.9.3 [DCT/DRAM Initialization and Resume].

Table 23: DCT training specific register values

Register	Value before	Value after
D18F2x78[AddrCmdTriEn]	0	IF (D18F2x94[MemClkFreq] <= 533 MHz) THEN 1 ELSE 0 ENDIF.
D18F2x8C[DisAutoRefresh]	1	0
D18F2x90[DbeSkidBufDis]	IF (D18F2xF4_x40[Trcd] at target MEMCLK frequency > 0101b) THEN 0 ELSE 1 ENDIF.	
D18F2x94[BankSwizzleMode]	0	1
D18F2x94[PowerDownEn]	0	1
D18F2x94[ZqcsInterval]	00b	10b
D18F2x9C_x0000_000D[RxMaxDurDllNoLock]	0h	See 2.9.3.8.
D18F2x9C_x0000_000D[TxMaxDurDllNoLock]	0h	See 2.9.3.8.
D18F2x9C_x0D0F_0[F,7:0]10[EnRxPadStandby]	0	See 2.9.3.8.
D18F2xA4[ThrottleEn]	00b	See 2.9.7
D18F2xA8[BankSwap]	0	0
D18F2xA8[DbeGskMemClkAlignMode]	00b	See 2.9.3
D18F2xF4_x32[DataTxFifoSchedDlyNegSlot1, DataTxFifoSchedDlySlot1, DataTxFifoSchedDlyNegSlot0, DataTxFifoSchedDlySlot0]	See 2.9.3.2.2.	See 2.9.3.2.2
D18F2x11C[PrefCpuDis]	1	0
D18F2x11C[DctWrLimit]	1Fh	1Ch
D18F2x1C0[DramTrainPdbDis]	0	1
D18F3x7C	See Table 8	See Table 8
D18F3x17C	See Table 8	See Table 8
D18F3x188[EnCpuSerRdBehindNpIoWr]	1	0
D18F4x1A8[DramSrEn]	0	1
MSRC001_0015[ForceUsRdWrSzPrb]	0	0
D18F6x78[DbeCmdThrottle]	00h	00h

1. Programmed specific to the current memory configuration.

2.9.3.6 DRAM Device Initialization

BIOS initializes the DRAM devices and the controller using a software controlled sequence. See 2.9.3.6.1 [Software DDR3 Device Initialization].

DRAM initialization is complete after D18F2x7C[EnDramInit] is written by BIOS from 1 to 0 in the software-controlled sequence.

See 2.9.3.5 [DCT Training Specific Configuration] for additional training requirements.

2.9.3.6.1 Software DDR3 Device Initialization

BIOS should use the following procedure to initialize the DDR3 DIMMs on the channel. This procedure should be run only when booting from an unpowered state (ACPI S4, S5 or G3; not S3, suspend to RAM).

See 2.9.3.6.1.1 [DDR3 MR Initialization].

This procedure assumes the DRAM channel frequency is set and D18F2x7C[EnDramInit] is programmed to 1 prior to executing the following steps. See 2.9.3.2 and 2.9.3.2.2.

1. Configure the DCT registers, including MemClkFreq and MemClkFreqVal.
2. Program D18F2x7C[EnDramInit] = 1.
3. Wait 200 us.
4. Program D18F2x7C[DeassertMemRstX] = 1.
5. Wait 500 us.
6. Program D18F2x7C[AssertCke] = 1.
7. Wait 360 ns.
8. Send MRS(2).
9. Send MRS(3). Ordinarily at this time, MrsAddress[2:0] = 000b.
10. Send MRS(1) with MrsAddress[7] = 0.
11. Send MRS(0) with MrsAddress[8] = 1.
12. Send a ZQCL command.
13. Program D18F2x7C[EnDramInit] = 0.

BIOS instructs the DCT to send a ZQCL command by programming D18F2x7C as follows:
1. Program MrsAddress[10] = 1.
2. Program SendZQCmd = 1.
3. Wait for SendZQCmd = 0.
4. Wait 512 MEMCLKs.

2.9.3.6.1.1 DDR3 MR Initialization

BIOS instructs the DCT to send MRS commands by programming D18F2x7C as follows:
1. Program MrsBank and MrsAddress as specified by Table 24, Table 25, Table 26, and Table 27:
   - MrsBank[2:0] = BA2:BA0.
   - MrsAddress[15:0] = A15:A0.
   - See D18F2x[4C:40][OnDimmMirror].
2. Program MrsChipSel as appropriate.
3. Program SendMrsCmd = 1.
4. Wait for SendMrsCmd = 0.

Table 24. DDR3 MR0

Address Field	Field	Value
BA2:BA0	MR Select	000b
A15:A13	Reserved	000b
A12	PPD	D18F2x84[PchgPDModeSel]
A11:A9	WR	D18F2x84[Twr]
A8	DLL	Controlled as required by the initialization sequence
A7	TM	0
A6:A4,A2	CAS Latency	{D18F2x88[Tcl[2:0]], D18F2x88[Tcl[3]]}
A3	RBT	1
A1:A0	BL	D18F2x84[BurstCtrl]

Table 25. DDR3 MR1

Address Field	Field	Value
BA2:BA0	MR Select	001b
A15:A13	Reserved	000b
A12	Qoff	0b
A11	TDQS	0
A10	Reserved	0b
A8	Reserved	0b
A7	Level	Controlled as required by the initialization sequence
A4:A3	AL	00b
A9, A6, A2	Rtt_Nom	See 2.9.3.4.5 [DRAM ODT Control]
A5, A1	DIC	01b
A0	DLL	0

Table 26. DDR3 MR2

Address Field	Field	Value
BA2:BA0	MR Select	010b
A15:A11	Reserved	0_0000b
A10:A9	Rtt_Wr	See 2.9.3.4.5 [DRAM ODT Control]
A8	Reserved	0b
A7	SRT	See ASR
A6	ASR	See DIMM SPD Byte 31: SDRAM Thermal and Refresh Options
A5:A3	CWL	D18F2x84[Tcwl]
A2:A0	PASR	000b

Table 27. DDR3 MR3

Address Field	Field	Value
BA2:BA0	MR Select	011b
A15:A3	Reserved	000_0000_0000b
A2	MPR	0b
A1:A0	MPR Loc	00b

2.9.3.7 DRAM Training

This section describes the recommended methods used to train the processor DDR interface to DRAM for optimal functionality and performance. DRAM training is performed by BIOS after initializing the DRAM controller. See 2.9.3.6 [DRAM Device Initialization].

Some of the DRAM training steps described in this section require two passes if the target MEMCLK frequency is greater than the lowest supported MEMCLK frequency. For optimal software performance, software may defer the second pass (at target MEMCLK frequency) for each training step until after the first pass (at lowest supported frequency) of all other training steps are complete. See D18F2x94[MemClkFreq].

See 2.9.3.5 [DCT Training Specific Configuration] for additional training requirements.

See 2.5.1.3 [BIOS Requirements for Power Plane Initialization] for power plane initialization to be performed before and after DRAM training.

In the following subsections, lane is used to describe an 8-bit wide data group, each with its own timing control.

2.9.3.7.1 Write Levelization Training

Write levelization involves using the phy to detect the edge of DQS with respect to the memory clock on the DIMM for write accesses to each lane.

Training is accomplished on a per DIMM basis. If the target frequency is greater than the lowest supported MEMCLK frequency then BIOS performs multiple passes; otherwise, only one pass is required. See 2.9.3.2.2.1 [Requirements for DRAM Frequency Change During Training].

- Pass 1: Configure the memory subsystem for the lowest supported MEMCLK frequency. See D18F2x94[MemClkFreq].
- Pass 2 - Pass N: Configure the memory subsystem for the next higher supported MEMCLK frequency. Repeat until the target MEMCLK frequency is reached.

The following describes the steps used for each pass of write levelization training:

For each DIMM:

1. Prepare the DIMMs for write levelization using DDR3-defined MR commands. See 2.9.3.6.1.1 [DDR3 MR Initialization].
   A. Configure the output driver and on-die termination of the target DIMM as follows:
      - For the first rank of the target DIMM, enable write leveling mode and enable the output driver.
      - For all other ranks of the target DIMM, enable write leveling mode and disable the output driver.
      - For two or more DIMMs per channel, program Rtt_Nom of the target rank to the corresponding

specified Rtt_Wr termination. Otherwise, configure Rtt_Nom of the target DIMM as normal. See 2.9.3.4.5 [DRAM ODT Control].

B. Configure Rtt_Nom on the non-target DIMMs as normal. See 2.9.3.4.5.

2. Wait 40 MEMCLKs.
3. Configure the phy for write levelization training:
   A. Program D18F2x9C_x0000_0008[WrtLvTrEn]=0.
   B. Program D18F2x9C_x0000_0008[TrDimmSel] to specify the target DIMM to be trained.
   C. Program D18F2x9C_x0000_0008[WrLvOdt[3:0]] to the proper ODT settings for the current memory subsystem configuration. See 2.9.3.4.5 and Table 20.
   D. Program D18F2x9C_x0000_0008[WrLvOdtEn]=1.
   E. MFENCE.
   F. Wait 10 MEMCLKs to allow for ODT signal settling.
   G. For each lane program an initial value to registers D18F2x9C_x0000_00[51:50] to set the gross and fine delay. See 2.9.3.7.1.1 [Write Levelization Seed Value].
4. Perform write leveling of the devices on the DIMM:
   A. Program D18F2x9C_x0000_0008[WrtLvTrEn]=1.
   B. MFENCE.
   C. Wait 200 MEMCLKs.
   D. Program D18F2x9C_x0000_0008[WrtLvTrEn]=0.
   E. Read from registers D18F2x9C_x0000_00[51:50] to get the gross and fine delay settings for the target DIMM and save these values.
5. Disable write levelization training so that the phy stops driving write levelization ODT.
   A. Program D18F2x9C_x0000_0008[WrLvOdtEn]=0.
   B. MFENCE.
   C. Wait 10 MEMCLKs to allow for ODT signal settling.
6. Program the target DIMM back to normal operation by configuring the following (see step 2 above):
   - Configure all ranks of the target DIMM for normal operation.
   - Enable the output drivers of all ranks of the target DIMM.
   - For a two or more DIMM system, program the Rtt_Nom value for the target DIMM to the normal operating termination.

- Calculate and program the final saved gross and fine delay values into D18F2x9C_x0000_00[44:30] [DRAM DQS Write Timing Control].
   A. GrossDly = SeedGross + PhRecGrossDlyByte - SeedPreGross.
   B. IF (GrossDly < 0) THEN
      Program WrDqsFineDly = 0
      Program WrDqsGrossDly = 0
      ELSE
      Program WrDqsFineDly = PhRecFineDlyByte
      Program WrDqsGrossDly = GrossDly
      ENDIF.

2.9.3.7.1.1 Write Levelization Seed Value

The seed value for pass 1 of write levelization training is design and platform specific and should be determined by characterization for best performance. The seed delay value must fall within +/- 1.20 ns, including PVT and jitter, of the measured clock delay.

The following steps are taken to determine the seed values needed to program the DRAM Phase Recovery Control registers:

For each pass:

1. Calculate the total seed based on the following:
   - Pass 1: IF SO-DIMM THEN SeedTotal = 12h ELSE SeedTotal = 1Ah. ENDIF.
   - Pass 2 - Pass N:
     - SeedTotalPreScaling = the total delay values in D18F2x9C_x0000_00[44:30] from the previous pass of write levelization training.
     - SeedTotal = FLOOR(SeedTotalPreScaling*(target frequency)/(frequency from previous pass)).
2. SeedGross = SeedTotal DIV 32.
3. SeedFine = SeedTotal MOD 32.
4. If (SeedGross is odd)
   then SeedPreGross = 1
   else SeedPreGross = 2.
5. Program D18F2x9C_x0000_00[51:50][PhRecFineDlyByte] = SeedFine.
6. Program D18F2x9C_x0000_00[51:50][PhRecGrossDlyByte] = SeedPreGross.

2.9.3.7.2 DQS Receiver Enable Training

Receiver enable delay training is used to dynamically determine the optimal delay value for D18F2x9C_x0000_00[24:10] [DRAM DQS Receiver Enable Timing Control]. The optimal DQS receiver enable delay value is platform and load specific, and occurs in the middle of a received read preamble. The timing of the preamble includes the inbound DQS propagation delay, which is unknown by BIOS. The training for delay values involves:

1. Configuring the phy for an initial expected phase value (seed).
2. Generating a stream of read DQS edges from the DRAM by issuing multiple read commands.
3. The phy determining the phase between the received DQS edges and a reference clock.
4. Calculating a final delay value for enabling receivers during normal read operations using the phase determined by the phy.

Training is accomplished on a per channel, per DIMM, per rank basis. If the target frequency is greater than the lowest supported MEMCLK frequency then BIOS performs multiple passes; otherwise, only one pass is required. See 2.9.3.2.2.1 [Requirements for DRAM Frequency Change During Training].

- Pass 1: Configure the memory subsystem for the lowest supported MEMCLK frequency. See D18F2x94[MemClkFreq].
- Pass 2 - Pass N: Configure the memory subsystem for the next higher supported MEMCLK frequency. Repeat until the target MEMCLK frequency is reached.

The following describes the steps used for each pass of receiver enable training:

- Program D18F2x78[MaxRdLatency] = 12h.

For each rank:

- Ensure that all ranks of the DIMM are configured for burst length 8 mode.

1. Program D18F2x9C_x0000_0008[TrDimmSel] to specify the target DIMM to be trained.
2. For each lane program an initial value to registers D18F2x9C_x0000_00[51:50] and D18F2x9C_x0000_00[24:10] to set the gross and fine delay as specified in 2.9.3.7.2.1 [DQS Receiver Enable Training Seed Value].
3. Program D18F2x9C_x0000_0008[DqsRcvTrEn]=1.
4. Issue 192 read requests to the target rank by issuing three sets of 64 read requests each. For each set of 64,

the reads must be to consecutive DRAM column addresses (i.e. 64 bytes apart) and must not cross a naturally aligned 4 KB boundary. To generate the needed continuous read streams for training, see 2.9.3.7.5 [DRAM Training Pattern Generation].
5. Program D18F2x9C_x0000_0008[DqsRcvTrEn]=0.
6. Read D18F2x9C_x0000_00[51:50][PhRecGrossDlyByte, PhRecFineDlyByte] to get the gross and fine delay values for each lane.
7. For each lane, calculate and program the corresponding receiver enable delay values for D18F2x9C_x0000_00[24:10][DqsRcvEnGrossDelay, DqsRcvEnFineDelay]. Save the result for use later.
   - DqsRcvEnFineDelay = PhRecFineDlyByte
   - DqsRcvEnGrossDelay = SeedGross + PhRecGrossDlyByte - SeedPreGross +1.

- For each rank pair on a dual-rank DIMM, compute the average value of the total delays saved during the training of each rank and program the result in D18F2x9C_x0000_00[24:10][DqsRcvEnGrossDelay, DqsRcvEnFineDelay].

2.9.3.7.2.1 DQS Receiver Enable Training Seed Value

The seed value for pass 1 of receiver enable delay training is design and platform specific and must be determined by characterization. The seed value represents the total delay from a reference point to the left edge of the read preamble on a read CAS measured at the processor pins, in 1 UI/32 increments. The reference point is defined as the clock in which CAS is asserted + CL - 1. This value is expected to be larger than 2 UI in the steps below. The phy adds a fixed offset to the delay seed value prior to sampling read DQS edges.

The following steps are taken to determine the seed values needed to program the DRAM Phase Recovery Control registers:

For each pass and each lane:

1. Calculate the total seed based on the following:
   - Pass 1: SeedTotal = HW_RXEN_SEED + the total delay values obtained from the first pass of write levelization training. See 2.9.3.7.1 [Write Levelization Training]. BIOS provides a default value of HW_RXEN_SEED=5Bh based on the characterization of AMD's reference platforms. System BIOS overrides this value with a different offset if characterization determined that 5Bh is not optimal for the target platform.
   - Pass 2 - Pass N:
     - SeedTotalPreScaling = (the total delay values in D18F2x9C_x0000_00[24:10] from the previous pass of DQS receiver enable training) - 20h.
     - SeedTotal = FLOOR(SeedTotalPreScaling*(target frequency)/(frequency from previous pass)).
2. SeedGross = SeedTotal DIV 32.
3. SeedFine = SeedTotal MOD 32.
4. If (SeedGross is odd)
   then SeedPreGross = 1
   else SeedPreGross = 2.
5. Program D18F2x9C_x0000_00[51:50][PhRecFineDlyByte] = SeedFine.
6. Program D18F2x9C_x0000_00[51:50][PhRecGrossDlyByte] = SeedPreGross.
7. Program D18F2x9C_x0000_00[24:10][DqsRcvEnGrossDelay] = SeedGross.

2.9.3.7.3 DQS Position Training

DQS position training is used to place the DQS strobe in the center of the read DQ data eye and to center the write DQ data eye across the write DQS strobe. Determining the correct DRAM DQS and DQ delay settings for both reads and writes is done by performing a two dimensional search of the delay settings found in

D18F2x9C_x0000_0[1:0]0[6:5] [DRAM Read DQS Timing Control] and D18F2x9C_x0000_0[1:0]0[2:1] [DRAM Write Data Timing].

Training is accomplished on a per channel, per rank, and per lane basis. BIOS uses the mutual passing delay values of each rank of a dual rank DIMM to calculate the optimal delay values.

For DQS position training, BIOS generates a training pattern using continuous read or write data streams. See 2.9.3.7.5 [DRAM Training Pattern Generation]. A 256-bit-time training pattern is recommended for optimal results.

Prior to DQS position training, BIOS must program D18F2x78[MaxRdLatency] based on the current greatest value of D18F2x9C_x0000_00[24:10]. See 2.9.3.7.4 [Calculating MaxRdLatency].

The following describes the steps used for DQS position training:

For each rank and lane:

1. Select a 64 byte aligned test address.
2. For each write data delay value in D18F2x9C_x0000_0[1:0]0[2:1] from Wr-DQS to Wr-DQS plus 1 UI, using the Wr-DQS delay value found in 2.9.3.7.1 [Write Levelization Training]:
   A. Program the write data delay value for the current lane.
   B. Write the DRAM training pattern to the test address.
   C. For each read DQS delay value in D18F2x9C_x0000_0[1:0]0[6:5] from 0 to 1 UI:
      a. Program the read DQS delay value for the current lane.
      b. Read the DRAM training pattern from the test address.
      c. Record the result for the current settings as a pass or fail depending if the pattern is read correctly.
      d. Program D18F2x78[RxPtrInitReq]=1.
      e. Wait for D18F2x78[RxPtrInitReq]=0.
3. Process the array of results and determine the longest string of consecutive passing read DQS delay values.
   A. If the read DQS delay results for the current lane contain three or more consecutive passing delay values, then calculate RdDqsTimeTrained as the average value of the smallest and largest delay values in the string of consecutive passing results.
   B. IF (D18F2x94[MemClkFreq] >= 667 MHz)
      THEN program D18F2x9C_x0000_0[1:0]0[6:5][RdDqsTime] = MAX(10h, RdDqsTimeTrained)
      ELSE program D18F2x9C_x0000_0[1:0]0[6:5][RdDqsTime] = RdDqsTimeTrained
      ENDIF.
4. Process the array of results and determine the longest string of consecutive passing write data delay values for the read DQS delay value found in the step above.
   - If the write data delay results for the current lane contain three or more consecutive passing delay values, then program D18F2x9C_x0000_0[1:0]0[2:1] with the average value of the smallest and largest delay values in the string of consecutive passing results.

See Figure 7.

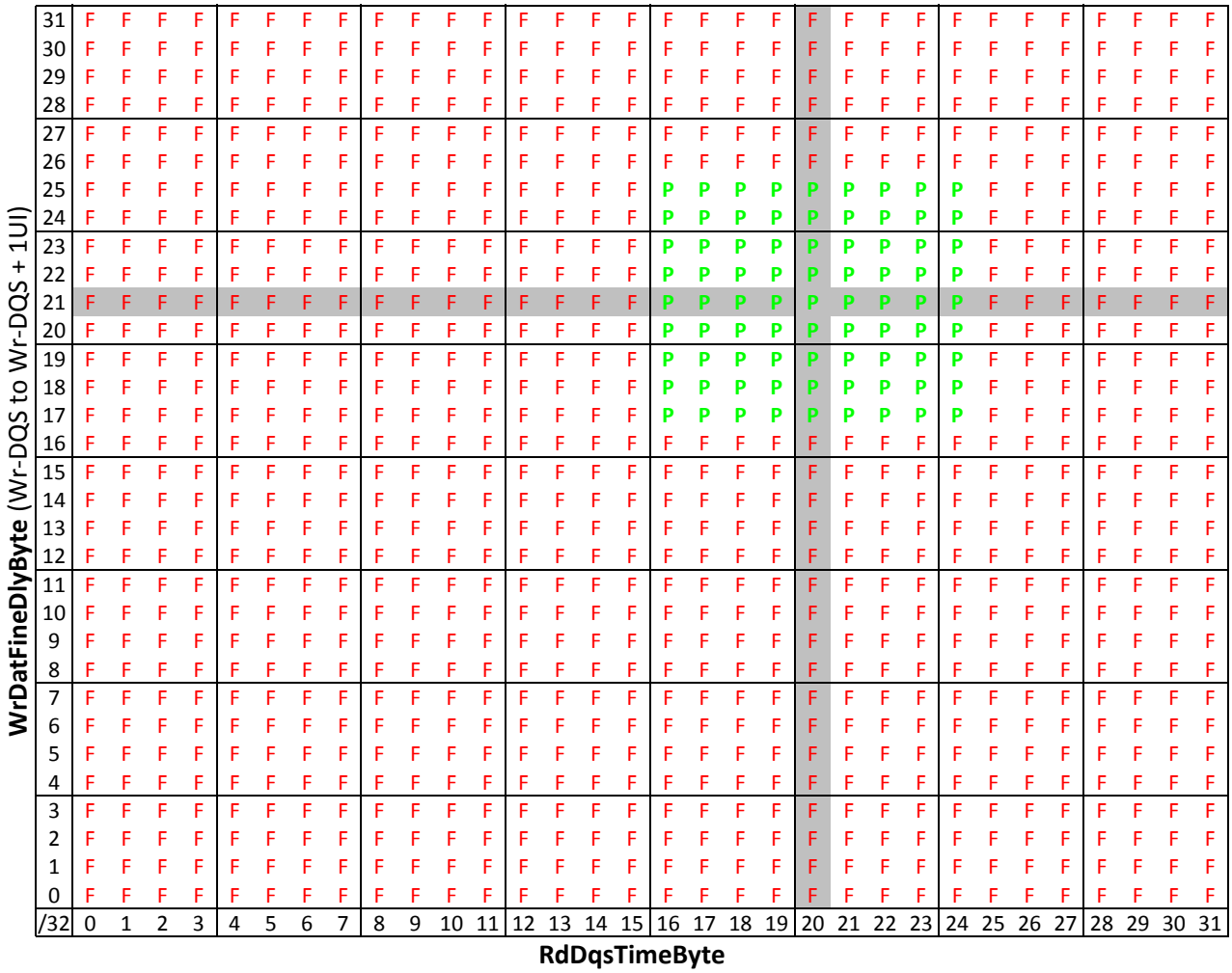
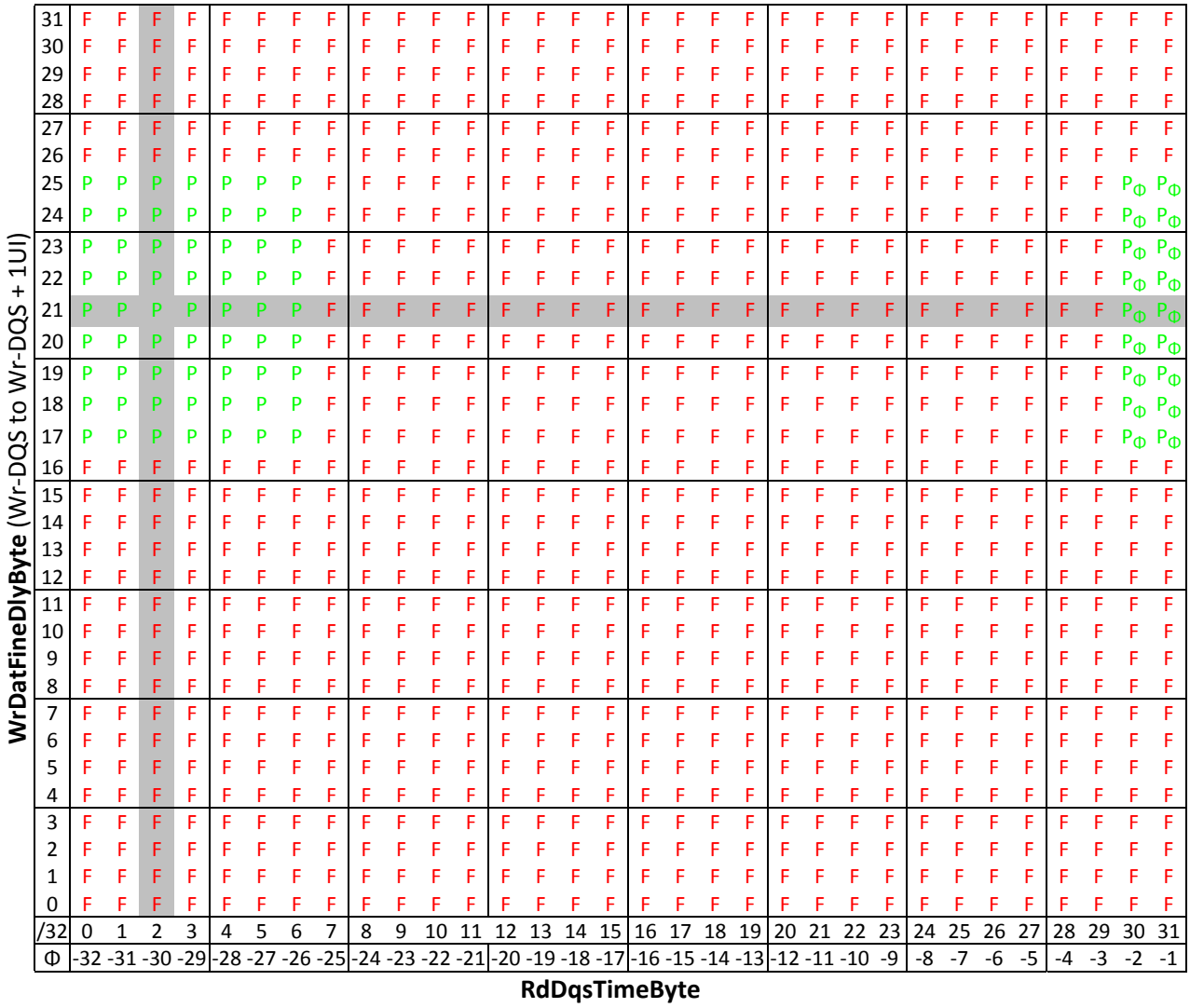


Figure 7: DQS Position Training Example Results

In some cases, a non-zero process, voltage, and temperature dependent insertion delay is added to the DLL programmed read DQS delay. This has the effect of sampling data later than intended and can result in missing the left edge of the passing region when sweeping from 0 to 1 UI because a read DQS delay value of 0 is already in the passing region. Since DQS is periodic, BIOS can recover the missing information by adjusting the algorithm described above to analyze both the in phase data and the data shifted by one bit time at each step of the read DQS delay sweep. See D18F2x1E8[TrainCmpSts2, TrainCmpSts].

As shown in Figure 8, for each delay setting BIOS records a passing result of P_Φ for the data comparison shifted by one bit time if the data at bit times N=0, 1, …, 6, is read correctly when compared against the data written at bit times N=1, 2, …, 7. In the array of results, these passing values make up the left piece of information that had been lost due to insertion delay. In order to process the array of results, BIOS calculates the read DQS delay value for a P_Φ result as RdDqsTimeByte minus 1 UI.



31	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
30	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
29	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
28	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
27	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
26	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
25	P	P	P	P	P	P	P	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	P_Φ	P_Φ
24	P	P	P	P	P	P	P	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	P_Φ	P_Φ
23	P	P	P	P	P	P	P	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	P_Φ	P_Φ
22	P	P	P	P	P	P	P	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	P_Φ	P_Φ
21	P	P	P	P	P	P	P	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	P_Φ	P_Φ
20	P	P	P	P	P	P	P	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	P_Φ	P_Φ
19	P	P	P	P	P	P	P	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	P_Φ	P_Φ
18	P	P	P	P	P	P	P	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	P_Φ	P_Φ
17	P	P	P	P	P	P	P	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	P_Φ	P_Φ
16	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
15	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
14	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
13	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
12	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
11	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
10	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
9	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
8	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
7	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
6	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
5	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
4	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
3	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
2	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
1	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
0	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
/32	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Φ	-32	-31	-30	-29	-28	-27	-26	-25	-24	-23	-22	-21	-20	-19	-18	-17	-16	-15	-14	-13	-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1

Figure 8: DQS Position Training Insertion Delay Recovery Example Results

2.9.3.7.4 Calculating MaxRdLatency

The MaxRdLatency value determines when the processor can receive incoming data from the DCT. Calculating MaxRdLatency consists of summing all the synchronous and asynchronous delays in the path from the processor to the DRAM and back at a given MEMCLK frequency. BIOS incrementally calculates the MaxRdLatency and then finally programs the value into D18F2x78[MaxRdLatency].

The following steps describe the algorithm used to compute D18F2x78[MaxRdLatency] used for DRAM training. P, N, and T are used as temporary placeholders for the incrementally summed value.

P = N = T = 0.

1. Pre- Tx FIFO adjustments
   A. IF (D18F2x9C_x0000_0004[AddrCmdSetup]!= D18F2x9C_x0000_0004[CkeSetup])
      THEN P = P + 1
      ENDIF.
   B. IF (D18F2xA8[DbeGskMemClkAlignMode]==01b ||

(D18F2xA8[DbeGskMemClkAlignMode]==00b && !(D18F2x9C_x0000_0004[AddrCmdSetup]== D18F2x9C_x0000_0004[CsOdtSetup]== D18F2x9C_x0000_0004[CkeSetup])))
THEN P = P + 1
ENDIF.

C. IF (D18F2x94[SlowAccessMode]==1)
THEN P = P + 2
ENDIF.

2. P = P + PtrSeparation
   - PtrSeparation = ((16 + RdPtrInitMin - D18F2x78[RdPtrInit]) MOD 16)
   - IF (D18F2x94[MemClkFreq] >= 667 MHz)
     THEN RdPtrInitMin = 2
     ELSE RdPtrInitMin = 3
     ENDIF.
3. P = P + 2
4. IF (D18F2x9C_x0000_0004[AddrCmdSetup]==0 && D18F2x9C_x0000_0004[CsOdtSetup]==0 && D18F2x9C_x0000_0004[CkeSetup]==0)
   THEN P = P + 1
   ELSE P = P + 2
   ENDIF.
5. P = P + (2 * (D18F2x88[Tcl] clocks - 1))
6. P = P + CEIL(MAX (total delay in D18F2x9C_x0000_00[24:10] + D18F2x9C_x0000_0[1:0]0[6:5][RdDqsTime]))
   - Use maximum DqsRcvEn total delay plus RdDqsTime across all DIMMs and all byte lanes.
   - Prior to DQS position training, use maximum value for RdDqsTime.
7. IF (D18F2x78[DisCutThroughMode]==0)
   THEN P = P + 3
   ELSE P = P + 7
   ENDIF.
8. P = P + 6.5
9. T = T + 2586 ps
10. N = (P/(MemClkFreq * 2) + T) * NclkFreq
    - NclkFreq = NCLK frequency as defined by D18F3xDC[NbPs0NclkDiv] or D18F6x90[NbPs1NclkDiv] for given NB P-state.
    - MemClkFreq = MEMCLK frequency as defined by D18F2x94[MemClkFreq].
11. IF (D18F6x98[NbPs1Act]==0)
    THEN N = N + 2
    ELSE N = N + 1
    ENDIF.
12. D18F2x78[MaxRdLatency] = CEIL(N)

2.9.3.7.4.1 MaxRdLatency Training

After DRAM training, BIOS optimizes D18F2x78[MaxRdLatency] using the following algorithm:

For MaxRdLatency training, BIOS generates a training pattern using continuous read or write data streams. See 2.9.3.7.5 [DRAM Training Pattern Generation]. The following three cache line pattern is used to train the MaxRdLatency value:

0C3C_FF52_6E0E_3FACh
49C5_B613_4A68_8181h
5C16_50E3_7C78_0BA6h

0C67_53E6_0C4F_9D76h
BABF_B6CA_2055_35A5h
0C5F_1C87_610E_6E5Fh
14C9_C383_4884_93CEh
9CE8_F615_F5B9_A5CDh

C38F_1B4C_AAD7_14B5h
669F_7562_72ED_647Ch
4A89_8B30_5233_F802h
3326_B465_10A4_0617h
C807_E3D3_5538_6E04h
14B4_E63A_AB49_E193h
EA51_7C45_67DF_2495h
F814_0C51_7624_CE51h

B61D_D0C9_4824_BD23h
E8F3_807D_072B_CFBEh
25E3_0C47_919E_A373h
4DA8_0A5A_FEB1_2958h
792B_0076_E9A0_DDF8h
F025_B496_E81C_73DCh
8085_94FE_1DB7_E627h
655C_7783_8266_8268h

To perform MaxRdLatency training:

1. Program D18F2x78[ForceCasToSlot0]=1.
2. Calculate a starting MaxRdLatency delay value by executing the steps in section 2.9.3.7.4. Save this as CalcMaxRdLatency.
3. Select six linearly contiguous 64 byte aligned test addresses associated with the DIMM that has the worst case D18F2x9C_x0000_00[24:10] register setting.
4. Write the six DIMM test addresses by twice repeating the three cache line pattern given above.
5. Program D18F2x78[MaxRdLatency] = CalcMaxRdLatency.
6. For (i = 0; i < 100; i++):
   A. Read the DIMM test addresses.
   B. Compare the values read against the pattern written.
      - If pattern is read correctly then save current MaxRdLatency as TrainedMaxRdLatency and decrement D18F2x78[MaxRdLatency].
7. Program D18F2x78[MaxRdLatency] = TrainedMaxRdLatency + TrainingOffset.
   - IF ((NCLK!=MEMCLK) && (NCLK!=MEMCLK/2))
     THEN TrainingOffset = 3
     ELSE TrainingOffset = 2 ENDIF.
8. Program D18F2x78[ForceCasToSlot0]=0.

2.9.3.7.5 DRAM Training Pattern Generation

DRAM training relies on generating a sequence of reads or writes between the processor and DRAM such that worst case electrical interactions are created. This section describes how these sequences are generated. DRAM training pattern generation uses the read/write training buffer for data storage. During write pattern generation, the write training buffer is filled by software, and the contents are burst to the DRAM interface. Conversely for reads, data bursts from the DRAM interface are stored in the read training buffer. When read data bursts are stored in the read training buffer they are automatically compared against the data in the write

training buffer. Software accesses the compare results by reading D18F2x1E8[TrainCmpSts2,TrainCmpSts].

Two address modes are available for DRAM training pattern generation. For generating a continuous sequence of reads or writes to the same rank, continuous pattern generation mode is used. See 2.9.3.7.5.1 [Continuous Pattern Generation]. To generate sequences of accesses to up to four different ranks or DIMMs, alternative address mode is used. See 2.9.3.7.5.2 [Alternative Address Mode].

2.9.3.7.5.1 Continuous Pattern Generation

Continuous pattern generation mode uses a single linearly contiguous sequence of DRAM training addresses to read or write training patterns to a single rank. Addresses must not cross a naturally aligned 4 KB boundary.

Program the following to enable continuous pattern generation mode:

1. Program D18F2x1C0[AltAddrEn]=0.
2. Program {D18F2x1CC[TrainAddrPtr[39:38]], D18F2x1C8[TrainAddrPtr[37:6]]} to the DRAM training start address.

TrainAddrPtr[39:6] is incremented by hardware after every cache line transfer.

2.9.3.7.5.2 Alternative Address Mode

Alternative address mode uses multiple linearly contiguous sequences of DRAM training addresses to read or write training patterns to different DIMMs or ranks. In this mode, the address alternates between four address pointers, and the DRAM data stream may not be continuous across multiple address pointers. Each address pointer consists of a pointer field and an iteration field. When a training sequence is initiated (D18F2x1C0[RdTrainGo or WrTrainGo]=1) and alternative address mode is enabled (D18F2x1C0[AltAddrEn]=1), hardware sequences through the training address pointers transferring the number of cache lines specified for each pointer (TrainAddrPtrIt +1) until the number of cache lines specified by D18F2x1C0[TrainLength] has been transferred. The associated DRAM training address pointer is incremented by hardware after every cache line transfer.

Program the following to enable alternative address mode pattern generation:

1. Program D18F2x1C0[AltAddrEn]=1.
2. Program {D18F2x1CC[TrainAddrPtr[39:38]], D18F2x1C8[TrainAddrPtr[37:6]]} to the DRAM training start address.
3. Program {D18F2x1CC[AltAddr1Ptr[39:38]], D18F2x[1E0:1D8][AltAddr1Ptr[37:6]]}.
4. Program {D18F2x1CC[AltAddr2Ptr[39:38]], D18F2x[1E0:1D8][AltAddr2Ptr[37:6]]}.
5. Program {D18F2x1CC[AltAddr3Ptr[39:38]], D18F2x[1E0:1D8][AltAddr3Ptr[37:6]]}.

2.9.3.7.5.3 Read Pattern Generation

Perform the following steps to read a training pattern from DRAM:

The DCT requires certain features be disabled to achieve continuous patterns. See 2.9.3.5 [DCT Training Specific Configuration].

1. Program D18F2x1C0[RdDramTrainMode]=1.
2. Program D18F2x1C0[TrainLength] to the appropriate number of cache lines.
3. Program the DRAM training address as follows:
   - If continuous pattern generation mode is desired, see 2.9.3.7.5.1 [Continuous Pattern Generation].

- Else for alternative address mode, see 2.9.3.7.5.2 [Alternative Address Mode].

4. Program D18F2x1D0[WrTrainBufAddr]=000h.
5. Program D18F2x1C0[RdTrainGo]=1.
6. Wait for D18F2x1C0[RdTrainGo]=0.
7. Read D18F2x1E8[TrainCmpSts] and D18F2x1E8[TrainCmpSts2].
8. Program D18F2x1C0[RdDramTrainMode]=0.

2.9.3.7.5.4 Write Pattern Generation

Write DRAM training is accomplished using the write training buffer. Perform the following steps to initialize the write training buffer:

1. Program D18F2x1C0[WrDramTrainMode]=1.
2. Program D18F2x1C0[TrainLength] to the appropriate number of cache lines.
3. Program D18F2x1D0[WrTrainBufAddr]=000h. Successively write each doubleword of the training pattern to D18F2x1D4.

After initializing the write training buffer, perform the following steps to write the pattern to DRAM:

The DCT requires certain features be disabled to achieve continuous patterns. See 2.9.3.5 [DCT Training Specific Configuration].

1. Program D18F2x1D0[WrTrainBufAddr]=000h.
2. Program the DRAM training address as follows:
   - If continuous pattern generation mode is desired, see 2.9.3.7.5.1 [Continuous Pattern Generation].
   - Else for alternative address mode, see 2.9.3.7.5.2 [Alternative Address Mode].
3. Program D18F2x1C0[WrTrainGo]=1.
4. Wait for D18F2x1C0[WrTrainGo]=0.
5. Program D18F2x1C0[WrDramTrainMode]=0.
6. If training is not complete, program D18F2x1C0[WrDramTrainMode]=1 and go to step 4 to issue next set of training writes.

2.9.3.8 DRAM Phy Power Savings

To configure the phy for lower power consumption, the following steps are performed on the DRAM channel:

1. Program D18F2x88[MemClkDis] to disable unused MEMCLK pins.
2. Program D18F2x9C_x0D0F_2[1:0]30[PwrDn] = 1 for unused MEMCLK pairs.
3. Program D18F2x9C_x0000_000C[ODTTri, ChipSelTri] to disable unused pins.
4. Program D18F2x9C_x0D0F_0[F,7:0]13[DllDisEarlyU] = 1b.
5. Program D18F2x9C_x0D0F_0[F,7:0]13[DllDisEarlyL] = 1b.
6. Program D18F2x9C_x0D0F_0[F,7:0]13[RxDqsUDllPowerDown] = 1b.
7. Program D18F2x9C_x0D0F_812F[7, 5, 0] = {1b, 1b, 1b}.
8. Program D18F2x9C_x0D0F_0[F,7:0]10[EnRxPadStandby]= IF (D18F2x94[MemClkFreq] <= 800 MHz) THEN 1 ELSE 0 ENDIF.
9. Program D18F2x9C_x0000_000D as follows:
   - TxMaxDurDllNoLock/RxMaxDurDllNoLock = 7h.
   - TxCPUpdPeriod/RxCPUpdPeriod = 011b.
   - TxDLLWakeupTime/RxDLLWakeupTime = 11b.

2.9.4 Chip Select Interleaving

The processor supports chip select memory interleaving which involves interleaving between the DIMM ranks of a channel. This mode interleaves the physical address space over multiple DIMM ranks, as opposed to each DIMM owning a single consecutive contiguous address space. This is accomplished by using lower-order address bits to select between DIMMs. See D18F2x[4C:40].

The chip select memory interleaving mode has the following requirements:
- The number of chip selects interleaved is a power of two.
- The chip selects are the same size and type.

A BIOS algorithm for programming D18F2x[4C:40] [DRAM CS Base Address] and D18F2x[64:60] [DRAM CS Mask] in memory interleaving mode is as follows:

1. Program all DRAM CS Base Address and DRAM CS Mask registers using contiguous normalized address mapping.
2. For each enabled chip select, swap the corresponding D18F2x[4C:40][BaseAddr[36:27]] bits with the D18F2x[4C:40][BaseAddr[21:13]] bits as defined in Table 28.
3. For each enabled chip select, swap the corresponding D18F2x[64:60][AddrMask[36:27]] bits with the D18F2x[64:60][AddrMask[21:13]] bits as defined in Table 28.

Table 28. DDR3 swapped normalized address lines for CS interleaving

DIMM Address Map	Chip Select Size	Swapped Base Address and Address Mask bits	
		4 way CS interleaving	2 way CS interleaving
0001b	256-MB	[29:28] and [17:16]	[28] and [16]
0010b	512-MB	[30:29] and [17:16]	[29] and [16]
0101b	1-GB	[31:30] and [17:16]	[30] and [16]
0111b	2-GB	[32:31] and [17:16]	[31] and [16]
1010b	4-GB	[33:32] and [17:16]	[32] and [16]
1011b	8-GB	[34:33] and [18:17]	[33] and [17]
1. See D18F2x80 [DRAM Bank Address Mapping].			

The following is an example of interleaving a 64-bit interface to DDR3 DRAM. The DRAM memory consists of two 512 MB dual rank DDR3 DIMMs.

1. The register settings for contiguous memory mapping are:
   D18F2x80 = 0000_0011h // CS0/1 = 256 MB; CS2/3 = 256 MB
   D18F2x40 = 0000_0001h // 0 MB base
   D18F2x44 = 0010_0001h // 256 MB base = 0 MB + 256 MB
   D18F2x48 = 0020_0001h // 512 MB base = 256 MB + 256 MB
   D18F2x4C = 0030_0001h // 768 MB base = 512 MB + 256 MB
   D18F2x60 = 0008_3FE0h // CS0/CS1 = 256 MB
   D18F2x64 = 0008_3FE0h // CS2/CS3 = 256 MB
2. The base address bits to be swapped are defined in Table 28, 256MB chip select size, 4 way CS interleaving column. Base address bits [29:28] bits are defined with D18F2x[4C:40][BaseAddr[21:20]]. Base address bits [17:16] are defined with D18F2x[4C:40][BaseAddr[9:8]].
   D18F2x40 = 0000_0001h

D18F2x44 = 0000_0101h
D18F2x48 = 0000_0201h
D18F2x4C = 0000_0301h

3. The address mask bits to be swapped are the same as the base address bits defined in the previous step. Address mask bits [29:28] are defined with D18F2x[64:60][AddrMask[21:20]]. Address mask bits [17:16] are defined with D18F2x[64:60][AddrMask[9:8]].
   D18F2x60 = 0038_3CE0h
   D18F2x64 = 0038_3CE0h

2.9.5 Memory Hoisting

Memory hoisting is defined as reclaiming (relocating) the unusable DRAM space that would naturally reside in the MMIO hole just below the 4GB address level. This memory is repositioned above the 4GB level when the register that controls memory hoisting, D18F1xF0 [DRAM Hole Address], is properly configured.

The region of DRAM that is hoisted is defined to be from D18F1xF0[DramHoleBase] to the 4GB level. The memory hoisting offset field, D18F1xF0[DramHoleOffset], is programmed as follows:
DramHoleOffset[31:23] = {(100h - DramHoleBase[31:24],0b};

2.9.6 DRAM CC6/PC6 Storage

DRAM is used to hold the state information of cores entering the C6 power management state. As part of the system setup if CC6/PC6 is enabled, BIOS configures a special region of DRAM to hold the state information. In operation, hardware protects this region from general system accesses while allowing the cores access during C-state transitions. See 2.5.3.2.3.2 [Core C6 (CC6) State] and 2.5.3.2.3.4 [Package C6 (PC6) State].

The special DRAM storage region is defined to be a fixed 16 MB beginning at the DRAM base address specified by D18F4x12C[C6Base]. BIOS must configure the storage region at the top of the DRAM range, and adjust D18F1x44[DramLimit] downward accordingly. See Table 29.

Table 29. C6Base Programming Example

DRAM Populated	D18F1x40[DramBase], D18F1x44[DramLimit]	D18F4x12C[C6Base]
256 MB	0 MB, 240 MB - 1	240 MB

BIOS must write D18F4x12C[C6Base]=0 on platforms that do not enable CC6/PC6.

After finalizing the system DRAM configuration, BIOS must program D18F2x118[C6DramLock]=1.

2.9.7 DRAM On DIMM Thermal Management

The DCT can throttle commands and adjust the Tref refresh rate based on the state of the processor's M_EVENT_L pin. The recommended BIOS configuration is as follows:

- BIOS may enable DCT command throttling by programming D18F2xA4 [DRAM Controller Temperature Throttle] if the platform supports the M_EVENT_L pin.
  - The recommended usage is for this pin to be connected to one or more JEDEC defined on DIMM temperature sensors. The DIMM SPD indicates on DIMM temperature sensor support.

  - BIOS configures the temperature sensor(s) to assert the M_EVENT_L pin active low when the trip point is exceeded and de-assert M_EVENT_L when the temperature drops below the trip point minus the sensor defined hysteresis.
  - BIOS programs D18F2xA4[ThrottleEn] with the throttling mode to employ when the trip point has been exceeded.
- BIOS configures D18F2x8C[Tref] based on JEDEC defined temperature range options, as indicated by the DIMM SPD ROM. The two defined temperature ranges are normal (with a case temperature of 85 °C), and extended (with a case temperature of 95 °C).
  - If all DIMMs support the normal temperature range, or if normal and extended temperature range DIMMs are mixed, BIOS programs D18F2x8C[Tref] to 7.8 us and D18F2xA4[DoubleTrefRateEn] =1. BIOS configures the temperature sensor trip point for all DIMMs according to the 85 °C case temperature specification.
  - If all DIMMs support the extended temperature range, BIOS has two options:
    a. Follow the recommendation for normal temperature range DIMMs.
    b. Program D18F2x8C[Tref] to 3.9 us and configure the temperature sensor trip point for all DIMMs according to the 95 °C case temperature specification.
- At startup, BIOS determines if the DRAMs are hot before enabling a DCT and delays for an amount of time to allow the devices to cool under the influence of the thermal solution. This is accomplished by checking the temperature status in the temperature sensor of each DIMM.
- The latched status of the M_EVENT_L pin can be read by system software in D18F2xAC[MemTempHot].

The relationship between DRAM case temperatures, trip points, and M_EVENT_L pin sampling intervals is outlined as follows:
- The trip points for each DIMM are configured to the case temperature specification minus a guardband temperature for the DIMM, except in the case of mixed extended temperature DIMM types noted above.
- The temperature guardband is vendor defined and is used to account for sensor inaccuracy and platform thermal design.

2.10 Thermal Functions

Thermal functions HTC and THERMTRIP are intended to maintain processor temperature in a valid range by:

- Providing a signal to the external circuitry for system thermal management such as fan control
- Lowering power consumption by switching to lower-performance P-state, or
- Sending the processor to the THERMTRIP state to prevent physical damage.

The processor thermal-related circuitry includes (1) the temperature calculation circuit (TCC) for determining the temperature of the processor and (2) logic that uses the temperature from the TCC.

2.10.1 The Tctl Temperature Scale

Tctl is a processor temperature control value used for processor thermal management. Tctl is accessible through SB-TSI and D18F3xA4[CurTmp]. Tctl is a temperature on its own scale aligned to the processors cooling requirements. Therefore Tctl does not represent a temperature which could be measured on the die or the case of the processor. Instead, it specifies the processor temperature relative to the maximum operating temperature, Tctl_max. Tctl is defined as follows for all parts:

A: For Tctl = 0 to Tctl_max - 0.125: the temperature of the part is [Tctl_max - Tctl] under the maximum operating temperature.

B: For Tctl = Tctl_max to 255.875: the temperature of the part is [Tctl - Tctl_max] over the maximum operating

temperature. The processor may take corrective actions that affects performance, such as HTC, to support the return to Tctl range A.

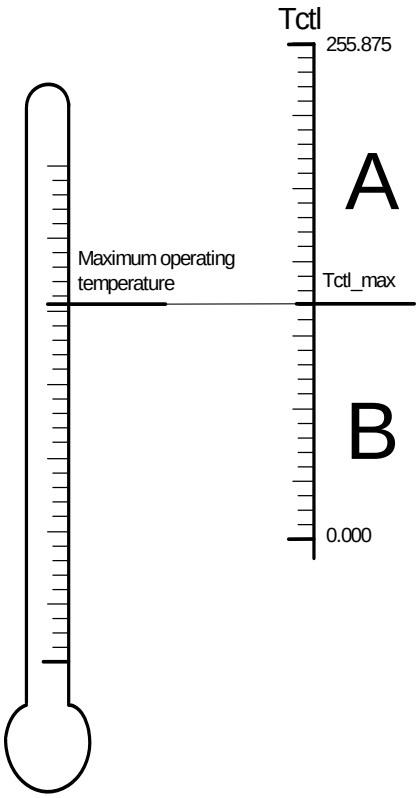


Figure 9: Tctl scale

2.10.2 Sideband Temperature Sensor Interface (SB-TSI)

The SB-TSI is used by an external SMBus master to access the internal temperature sensor and to specify temperature thresholds. The processor has access to the SB-TSI registers via D18F3x1E8 [SBI Address] and D18F3x1EC [SBI Data]. 100 kHz standard-mode and 400 kHz fast-mode are supported. 3.4 MHz high-speed mode is not supported.

2.10.3 Temperature-Driven Logic

The temperature calculated by the TCC is used by HTC, THERMTRIP, PROCHOT_L, and the serial interface, SB-TSI.

2.10.3.1 PROCHOT_L and Hardware Thermal Control (HTC)

The processor *HTC-active state* is characterized by (1) the assertion of PROCHOT_L, (2) reduced power consumption, and (3) reduced performance. While in the HTC-active state, the processor reduces power consumption by limiting all cores to a P-state specified by D18F3x64[HtcPstateLimit]. See 2.5.3.1 [Core P-states]. While in the HTC-active state, software should not change D18F3x64 except for HtcActSts or HtcEn. Any change to fields in D18F3x64 other than HtcActSts or HtcEn while in the HTC-active state can result in undefined behavior. HTC status and control is provided through D18F3x64.

The PROCHOT_L pin acts as both an input and as an open-drain output. As an output, PROCHOT_L is driven low to indicate that the HTC-active state has been entered due to an internal condition, as described by the fol-

lowing text.

The processor enters the HTC-active state if all of the following conditions are true:
- D18F3xE8[HtcCapable]=1.
- D18F3x64[HtcEn]=1.
- PWROK is asserted.
- THERMTRIP_L is de-asserted.
- The processor is not in the package C1 (PC1) state or package C6 (PC6) state.

and any of the following conditions are true:
- Tctl is greater than or equal to the HTC temperature limit (D18F3x64[HtcTmpLmt]).
- PROCHOT_L is asserted.

The processor exits the HTC-active state when all of the following are true:
- Tctl is less than the HTC temperature limit (D18F3x64[HtcTmpLmt]).
- Tctl has become less than the HTC temperature limit (D18F3x64[HtcTmpLmt]) minus the HTC hysteresis limit (D18F3x64[HtcHystLmt]) since being greater than or equal to the HTC temperature limit (D18F3x64[HtcTmpLmt]).
- PROCHOT_L is de-asserted.

If D18F3x64[HtcEn] is cleared from 1 to 0 while the processor is in the HTC-active state and PROCHOT_L is asserted, the processor remains in the HTC-active state until PROCHOT_L is de-asserted. Clearing D18F3x64[HtcEn] to 0 while PROCHOT_L is de-asserted causes the processor to exit the HTC-active regardless of the state of the other exit criteria.

The default value of the HTC temperature threshold (Tctl_max) is specified in the AMD Family 14h Processor Power and Thermal Datasheet.

2.10.3.2 THERMTRIP

If the processor supports the THERMTRIP state (as specified by D18F3xE4 [Thermtrip Status][ThermtpEn] or CPUID Fn8000_0007_EDX[TTP], which are the same) and the temperature approaches the point at which the processor may be damaged, the processor enters the THERMTRIP state. The THERMTRIP state is characterized as follows:

- The THERMTRIP_L signal is asserted.
- The RESET_L signal is not asserted.
- Nearly all clocks are gated off to reduce dynamic power.
- A low-value VID is generated.
- In addition, the external chipset is expected to place the system into the S5 ACPI state (power off) if THERMTRIP_L is detected to be asserted.

A cold reset is required to exit the THERMTRIP state.

2.11 Root complex

2.11.1 Overview

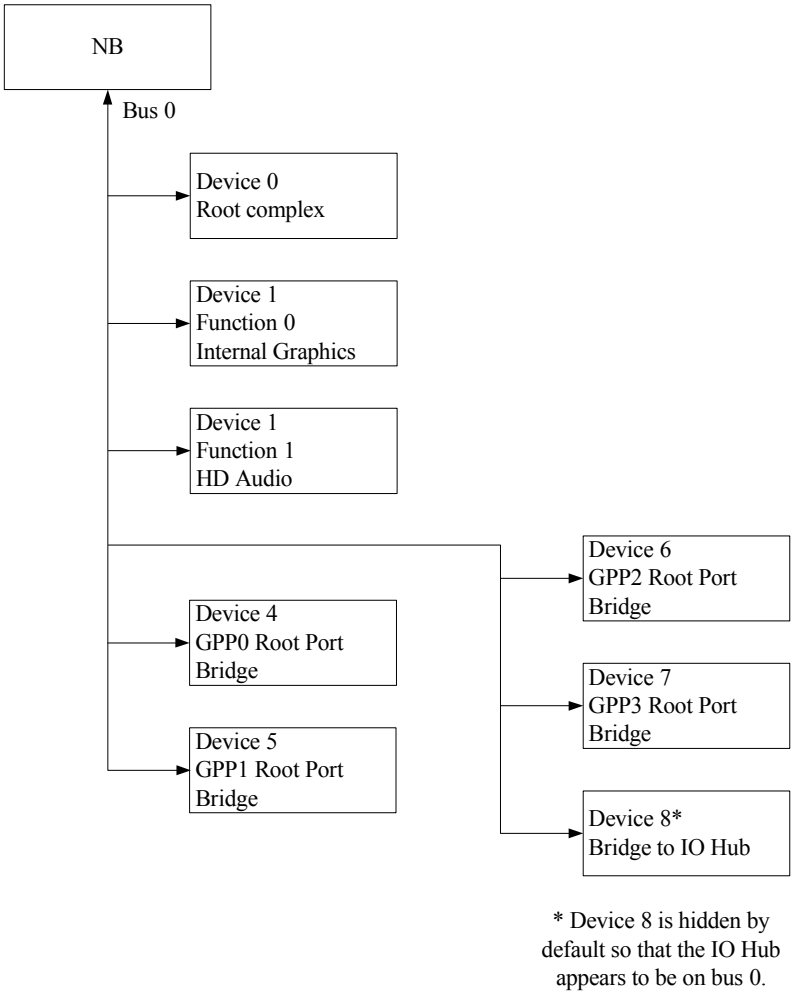


Figure 10: Root complex topology

2.11.2 Links

2.11.2.1 Overview

There are 5 configurable ports, consisting of 4 General Purpose Ports (GPP) and one x4 FCH port.

GPP links each have a Type 1 Virtual PCI-to-PCI bridge header in the PCI configuration space mapped to devices according to Figure 10.

Each PCIe lane is assigned a unique lane ID that software uses to communicate configuration information to the SMU. Table 30 shows the mappings between lane IDs and lanes.

Table 30: Lane ID Mapping

Lane ID	Lane
0	P_UMI_[T,R]X[P,N]0
1	P_UMI_[T,R]X[P,N]1
2	P_UMI_[T,R]X[P,N]2
3	P_UMI_[T,R]X[P,N]3
4	P_GPP_[T,R]X[P,N]0
5	P_GPP_[T,R]X[P,N]1
6	P_GPP_[T,R]X[P,N]2
7	P_GPP_[T,R]X[P,N]3

2.11.2.2 Link Configurations

The following link configurations are supported for the GPP links:

Table 31: Supported General Purpose (GPP) Link Configurations

D0F0xE4		GPP Lanes			
x0130_0080	x0110_0011	Lane 0	Lane 1	Lane 2	Lane 3
0000_0001h	0000_0300h	x4 Link			
0000_0002h	0000_010Ch	x2 Link		x2 Link	
0000_0003h	0000_0104h	x2 Link		x1 Link	x1 Link
0000_0004h	0000_0100h	x1 Link	x1 Link	x1 Link	x1 Link

2.11.3 Root Complex Configuration

2.11.3.1 LPC MMIO Requirements

To ensure proper operation of LPC generated DMA requests, the UMI must be configured to send processor generated MMIO writes that target the LPC bus to the FCH as non-posted writes. To ensure this requirement the MMIO address space of the LPC bus must not be included in the ranges specified by D18F1x[B8,B0,A8,A0,98,90,88,80] [Memory Mapped IO Base] and D18F1x[BC,B4,AC,A4,9C,94,8C,84] [Memory Mapped IO Limit] and non-posted protocol for memory writes must be enabled using the following sequence before LPC DMA transactions are initiated.

1. Configure the FCH to use the non-posted write protocol. See the FCH register specification for configuration details.
2. Program D0F0xE4_x0101_0010[UmiNpMemWrite] = 1.
3. Program D0F0x98_x06[UmiNpMemWrEn] = 1.

2.11.3.2 Miscellaneous Features

2.11.3.2.1 Straps

1. Program D0F0xE4_x0130_8011[StrapBifValid]=1.

2. Program strap values.
3. Program D0F0xE4_x0130_8011[StrapBifValid]=0.

2.11.3.2.2 Lane Reversal

Normally, the lanes of each port are physically numbered from n-1 to 0 where n is the number of lanes assigned to the port. Physical lane numbering can be reversed according to the following methods:

- To reverse the physical lane numbering for a specific port, program D[8:4]F0xE4_xC1[StrapReverseLanes]=1 according to the sequence in 2.11.3.2.1 [Straps].
- To reverse the physical lane numbering for all ports in the GPP or GFX interfaces, program D0F0xE4_x0101_00C0[StrapReverseAll]=1 according to the sequence in 2.11.3.2.1 [Straps].

Note that logical port numbering is established during link training regardless of the physical lane numbering.

2.11.3.2.3 Link Speed Changes

Link speed changes can only occur on Gen2 capable links. To verify that Gen2 speeds are supported verify D[8:4]F0x64[LinkSpeed]==02h.

2.11.3.2.4 De-emphasis

De-emphasis strength can be changed on a per-port basis by programming D[8:4]F0xE4_xB5[LcSelectDeemphasisCntl, LcSelectDeemphasis].

2.11.4 BIOS Timer

The root complex implements a 32-bit microsecond timer (see D0F0xE4_x0130_80F0 and D0F0xE4_x0130_80F1) that the BIOS can use to accurately time wait operations between initialization steps. To ensure that BIOS waits a minimum number of microseconds between steps BIOS should always wait for one microsecond more than the required minimum wait time.

2.12 System Management Unit (SMU)

The system management unit (SMU) is a subcomponent of the northbridge that is responsible for a variety of system and power management tasks during boot and runtime. Several internal registers are used to control various tasks. See 3.17 [GPU Memory Mapped Registers] for registers descriptions and details.

2.12.1 Microcontroller

The SMU contains a microcontroller with a 16k ROM and a 16k RAM.

2.12.1.1 Software Interrupts

BIOS or ACPI methods can interrupt the microcontroller to make firmware perform a specific task using to following sequence:

1. Wait for SMUx03[IntDone]==1.
2. If SMUx03[IntReq]==1, program SMUx03[IntReq]=0.
3. Program SMUx03[ServiceIndex] and set SMUx03[IntReq]=1. This may be done as a single write.
4. Wait for SMUx03[IntAck]==1.

After performing the steps above, software may continue execution before the interrupt has been serviced (see

SMUx03[IntDone]). However, software should not rely on the results of the interrupt until the service is complete. Interrupting the SMU with a service index that does not exist results in undefined behavior.

The available services indices are shown in Table 32.

Table 32: Microcontroller Interrupt Service Indices

Service Index	Notes
0Bh	Input: See 3.16 [Fixed Configuration Space (FCR)].
	Output: See 3.16 [Fixed Configuration Space (FCR)].
	Description: Reads or writes a register from FCR space.
	Firmware revision: 1.4
0Dh	Input: See 3.16 [Fixed Configuration Space (FCR)].
	Output: See 3.16 [Fixed Configuration Space (FCR)].
	Description: Reads a register from FCR space.
	Firmware revision: 1.4

2.13 Digital Display Interface

The processor contains a dedicated 2 port, 8 lane digital display interface (DDI) capable of generating two simultaneous independent display outputs from the internal GPU. The supported interface formats are shown in Table 33.

Table 33: Supported digital display interface formats

Format	Bit Rate	DDI Port 0 (LTDP0_TX[P,N][3:0])	DDI Port 1 (TDP1_TX[P,N][3:0])
LVDS	175 Mb/s to 805 Mb/s	Supported	Not supported
DisplayPort	2.7 Gb/s	Supported	Supported
DisplayPort	1.62 Gb/s	Supported	Supported
HDMI™/DVI Single Link Coherent	250 Mb/s to 1.65 Gb/s	Supported	Supported
HDMI™/DVI Single Link Incoherent	250 Mb/s to 1.65 Gb/s	Supported	Supported

There are two clock modes available for HDMI and DVI:
- Coherent mode uses a single PLL to provide a low jitter and low power clock source.
- Incoherent mode uses two cascaded PLLs to ensure that the data path jitter matches the clock path jitter.

The digital display interfaces are initialized and configured by the video BIOS and graphics driver.

2.13.1 DDI Clocking

Each digital display interface port contains an independent PLL to generate display clocks. Depending on the display interface format, the PLL uses either DISP_CLKIN_H/L or CLKIN_H/L as the reference clock. When DISP_CLKIN_H/L is the reference clock, the PLL can add programmable spread spectrum modulation to the display output. When CLKIN_H/L is the reference clock, the display output contains the same spread spectrum modulation as CLKIN_H/L. See the *Electrical Data Sheet for AMD Family 14h Models 00h-0Fh Processors, #44446*, for reference clock specifications.

Table 34: Digital display interface reference clock by format

Format	Reference Clock	Spread Spectrum Modulation
DisplayPort	CLKIN_H/L	From CLKIN_H/L
DVI	DISP_CLKIN_H/L	Programmable[1]
LVDS	DISP_CLKIN_H/L	Programmable[1]
HDMI	DISP_CLKIN_H/L	Programmable[1]

1. Spread spectrum modulation is controlled by the video BIOS and graphics driver.

2.14 Analog Display Interface

The processor contains an integrated analog display interface capable of driving an analog display directly from the internal GPU. This analog display interface contains three 10-bit video D/A converters for the RGB color signals. The analog display interface supports a maximum pixel frequency of 400 MHz, dynamic monitor detection (hot-plug), and power management modes including sleep and power-off.

The analog display interface is initialized and configured by the video BIOS and graphics driver.

2.15 Graphics Processor (GPU)

The processor contains an integrated DX11 compliant graphics processor.

2.15.1 GPU PCI Interface

BIOS must configure the PCI interface block of the GPU before the GPU can be accessed. The PCI interface block is configured by programming D0F0x64_x1C [Internal Graphics PCI Control 1] register. Two writes to this register are required. The first write sets the configuration and must have D0F0x64_x1C[WriteDis]=0. The second write uses the same write data with D0F0x64_x1C[WriteDis]=1.

2.15.2 Graphics Memory Controller (GMC)

The graphics memory controller is responsible for servicing memory requests from the different blocks within the GPU and forwarding routing them to the appropriate interface. The GMC is also responsible for translating GPU virtual address to GPU physical addresses and for translating GPU physical addresses to system addresses.

2.15.2.1 Register Save/Restore Engine (RENG)

The register save/restore engine (RENG) is used to save and restore GMC register state when power gating the GMC. The RENG uses a programmable RAM to specify the save/restore registers and to store the register data.

2.15.3 Frame Buffer (FB)

The frame buffer is defined as the portion of system memory dedicated for GPU use. BIOS should reserve at

least 64 MB of system memory for the frame buffer.

Table 35: BIOS recommended frame buffer sizes

Installed System Memory	Frame Buffer Size
< 1GB	64 MB
1GB <= memory < 2GB	256 MB
>= 2GB	384 MB

2.16 Machine Check Architecture

The processor contains logic and registers to detect, log, and (if possible) correct errors in the data or control paths in each core and the Northbridge.

Refer to the AMD64 Architecture Programmer's Manual for an architectural overview and methods for determining the processor's level of MCA support. See 1.2 [Reference Documents].

2.16.1 Machine Check Registers

CPUID Fn0000_0001_EDX[MCA] or CPUID Fn8000_0001_EDX[MCA] indicates the presence of the following machine check registers:

- MSR0000_0179 [Global Machine Check Capabilities (MCG_CAP)]
  - Reports how many machine check register banks are supported.
- MSR0000_017A [Global Machine Check Status (MCG_STAT)]
- MSR0000_017B [Global Machine Check Exception Reporting Control (MCG_CTL)]

The ability of hardware to generate a machine check exception upon an error is indicated by CPUID Fn0000_0001_EDX[MCE] or CPUID Fn8000_0001_EDX[MCE].

Once system software has determined that machine check registers exist via the CPUID instruction, MSR0000_0179 may be read to determine how many machine check banks are implemented and if MSR0000_017B [Global Machine Check Exception Reporting Control (MCG_CTL)] is present.

The error-reporting machine check register banks supported are:

- **MC0**: Data cache (DC).
- **MC1**: Instruction cache (IC).
- **MC2**: Bus unit (BU).
- **MC3**: Reserved.
- **MC4**: Northbridge (NB) including the IO link. These MSRs are also accessible from configuration space. There is only one NB error-reporting bank, independent of the number of cores.
- **MC5**: Fixed-issue reorder buffer (FR) machine check registers.

- The register types within each bank are:
  - **MCi_CTL**, Machine Check Control: Enables error reporting via machine check exception. The MC*i*_CTL register in each bank must be enabled by the corresponding enable bit in MCG_CTL (MSR0000_017B).
  - **MCi_STATUS**, Machine Check Status: Logs information associated with errors.
  - **MCi_ADDR**, Machine Check Address: Logs address information associated with errors.
  - **MCi_MISC**, Machine Check Miscellaneous: Log miscellaneous information associated with errors, as defined by each error type.
  - **MCi_CTL_MASK**, Machine Check Control Mask: Inhibit detection of an error source.

Table 36 identifies the registers associated with each error-reporting machine check register bank:

Table 36: MCA register cross-reference table

Register Bank (MC*i*)	MCA Register				
	CTL	STATUS	ADDR	MISC	CTL_MASK
MC0	MSR0000_0400	MSR0000_0401	MSR0000_0402	MSR0000_0403	MSRC001_0044
MC1	MSR0000_0404	MSR0000_0405	MSR0000_0406	MSR0000_0407	MSRC001_0045
MC2	MSR0000_0408	MSR0000_0409	MSR0000_040A	MSR0000_040B	MSRC001_0046
MC3	MSR0000_040C	MSR0000_040D	MSR0000_040E	MSR0000_040F	MSRC001_0047
MC4	MSR0000_0410	MSR0000_0411	MSR0000_0412	MSR0000_0413	MSRC001_0048
MC5	MSR0000_0414	MSR0000_0415	MSR0000_0416	MSR0000_0417	MSRC001_0049

Correctable and uncorrectable errors that are enabled in MCi_CTL are logged in MCi_STATUS and MCi_ADDR as they occur. Uncorrectable errors immediately result in a Machine Check exception.

Each MCi_CTL register must be enabled by the corresponding enable bit in MSR0000_017B [Global Machine Check Exception Reporting Control (MCG_CTL)].

Additionally, MSRC001_00[49:44] [Machine Check Control Mask (MCi_CTL_MASK)] allow BIOS to mask the presence of any error source enables from software for test and debug. When error sources are masked, it is as if the error was not detected. Such masking consequently prevents error responses.

Each register bank implements a number of machine check miscellaneous registers, denoted as MCi_MISCj, where j goes from 0 to a maximum of 8. The presence of valid information in the first MCi_MISC register (MCi_MISC0) is indicated by MCi_STATUS[MiscV], and in subsequent registers by MCi_MISCj[Valid]. If there is more than one MCi_MISC register in a given bank, a non-zero value in MCi_MISC0[BlkPtr] points to the contiguous block of additional registers.

2.16.2 Machine Check Errors

There are two classes of machine check errors defined:
- Correctable: errors that can be corrected by hardware or microcode and cause no loss of data or corruption of processor state.
- Uncorrectable: errors that cannot be corrected by hardware or microcode and may have caused the loss of data or corruption of processor state.

Correctable errors are always corrected (unless disabled by implementation-specific bits in control registers for test or debug reasons). If they are enabled for logging, the status and address registers in the corresponding register bank are written with information that identifies the source of the error.

Uncorrectable errors, if enabled for logging, update the status and address registers, and if enabled for reporting, cause a machine check exception. If there is information in the status and address registers from a previous correctable error, it is overwritten. If an uncorrectable error is not enabled for logging, the error is ignored.

The implications of the two main categories of errors are (shown with a non-exhaustive list of examples):

1. Corrected error; the problem was dealt with.

- Operationally (error handling), no action needs to be taken, because program flow is unaffected.
- Diagnostically (fault management), software may collect information to determine if any components should be de-configured or serviced.
- Examples include:
  - Correctable ECC, corrected online.

2. Uncorrected error; the problem was not dealt with.
   - Operationally (error handling), action does need to be taken, because program flow is affected.
   - Diagnostically (fault management), software may collect information to determine if and what components should be de-configured or serviced.
   - Examples include:
     - Uncorrectable ECC, no way to avoid passing it to process.

Machine check conditions can be simulated by using MSRC001_0015[McStatusWrEn]. This is useful for debugging machine check handlers.

2.16.2.1 Machine Check Error Logging and Reporting

An error is considered enabled for logging if:
- The global enable for the corresponding error-reporting bank in MSR0000_017B [Global Machine Check Exception Reporting Control (MCG_CTL)] is set to 1.
- The corresponding mask bit for the error in MSRC001_00[49:44] [Machine Check Control Mask (MCi_CTL_MASK)] is cleared to 0.

An error is considered enabled for reporting if:
- The error is enabled for logging.
- The corresponding enable bit for the error in MC*i*_CTL is set to 1.

Throughout the MCA register descriptions, the terms “enabled” and “disabled” generally refer to reporting, unless otherwise specified.

For details on error overflow, priority, and overwriting, see MCi_STATUS[Overflow].

Uncorrectable errors require software intervention. Therefore, when an uncorrectable error cannot be logged, critical error information may have been lost, and MCi_STATUS[PCC] may be set. If PCC is indicated, software should terminate system processing to prevent data corruption (see 2.16.2.4 [Handling Machine Check Exceptions]). If PCC is not indicated, there is no need to terminate the system, as any lost information was not critical.

2.16.2.2 Machine Check Error Logging Overwrite During Overflow

The MCi_STATUS[Over] bit indicates that an error was detected while the valid bit (Val) was set; at least one error was not logged. Overflow is set independently of whether the existing error is overwritten.

The following hierarchy identifies the error logging priorities.
1. Uncorrectable errors
2. Correctable errors

The machine check mechanism handles the contents of MC*i*_STATUS during overflow as follows:
- Higher priority errors overwrite lower priority errors.
- New errors of equal or lower priority do not overwrite existing errors.
- Uncorrectable errors which are not logged due to overflow result in setting PCC, unless the new uncorrect-

able error is of the same type and in the same reportable address range as the existing error.

During error overflow conditions (see MSR0000_0401[Over] and MSR0000_0411[Over]), an error which has already been logged in the status register may be overwritten.

Table 37 indicates which errors are overwritten in the MC0 and MC4 error status registers. Table 38 indicates which errors are overwritten in the MC1 and MC5 error status registers. Table 39 indicates which errors are overwritten in the MC2 error status register.

Table 37: MC0 and MC4 overwrite priorities

			Older Error			
			Uncorrectable		Correctable	
			Enabled	Disabled	Enabled	Disabled
Younger Error	Uncorrectable	Enabled	-	Overwrite	Overwrite	Overwrite
		Disabled	-	Overwrite	Overwrite	Overwrite
	Correctable	Enabled	-	Overwrite	Overwrite	Overwrite
		Disabled	-	Overwrite	Overwrite	Overwrite

Table 38: MC1 and MC5 overwrite priorities

			Older Error			
			Uncorrectable		Correctable	
			Enabled	Disabled	Enabled	Disabled
Younger Error	Uncorrectable	Enabled	-	Overwrite	Overwrite	Overwrite
		Disabled	-	Overwrite	Overwrite	Overwrite
	Correctable	Enabled	-	Overwrite	-	Overwrite
		Disabled	-	Overwrite	-	Overwrite

Table 39: MC2 overwrite priorities

			Older Error			
			Uncorrectable		Correctable	
			Enabled	Disabled	Enabled	Disabled
Newer Error	Uncorrectable	Enabled	-	-	Overwrite	Overwrite
		Disabled	-	-	Overwrite	Overwrite
	Correctable	Enabled	-	-	-	-
		Disabled	-	-	-	-

2.16.2.3 Machine Check Error Codes

The MCi_STATUS[ErrorCode] field contains information on the logged error. Table 40 identifies how to decode this field.

Three error code types are reported: TLB, memory, and bus errors. For a given error-reporting bank, Error Code Type is used in conjunction with the Extended Error Code (MCi_STATUS[ErrorCodeExt]) to uniquely identify the Error Type. Details for each Error Type are described in the tables accompanying the MCi_STATUS register for each bank.

- MC0 (DC); Table 106: [DC error signatures].
- MC1 (IC); Table 109: [IC error signatures].
- MC2 (BU); Table 111: [BU error signatures]
- MC4 (NB);Table 95 [NB error signatures, part 1] and Table 96 [NB error signatures, part 2].
- MC5 (FR); Table 114: [FR error signatures].

Table 40: Error code types

Error Code	Error Code Type	Description
`0000 0000 0001 TTLL`	TLB	TT = Transaction Type LL = Cache Level
`0000 0001 RRRR TTLL`	Memory	Errors in the cache hierarchy (not in NB) RRRR = Memory Transaction Type TT = Transaction Type LL = Cache Level
`0000 1PPT RRRR IILL`	Bus	General bus errors including link and DRAM PP = Participation Processor T = Timeout RRRR = Memory Transaction Type II = Memory or IO LL = Cache Level

Table 41: Error codes: transaction type (TT)

TT	Transaction Type
00	Instr: Instruction
01	Data
10	Gen: Generic
11	Reserved

Table 42: Error codes: cache level (LL)

LL	Cache Level
00	Reserved
01	L1: Level 1
10	L2: Level 2
11	LG: Generic

Table 43: Error codes: memory transaction type (RRRR)

RRRR	Memory Transaction Type
0000	Gen: Generic. Includes scrub errors.
0001	RD: Generic Read
0010	WR: Generic Write
0011	DRD: Data Read
0100	DWR: Data Write
0101	IRD: Instruction Fetch

Table 43: Error codes: memory transaction type (RRRR)

RRRR	Memory Transaction Type
0110	Prefetch
0111	Evict
1000	Snoop (Probe)

Table 44: Error codes: participation processor (PP)

PP	Participation Processor
00	SRC: Local node originated the request
01	RES: Local node responded to the request
10	OBS: Local node observed the error as a third party
11	Generic

Table 45: Error codes: memory or IO (II)

II	Memory or IO
00	Mem: Memory Access
01	Reserved
10	IO: IO Access
11	Gen: Generic

2.16.2.4 Handling Machine Check Exceptions

At a minimum, the machine check handler must be capable of logging errors for later examination. The handler should log as much information as is needed to diagnose the error.

More thorough exception handler implementations can analyze errors to determine if each error is recoverable. If a recoverable error is identified, the exception handler can attempt to correct the error and restart the interrupted program. Keep in mind that an error may not be recoverable for the process it directly affects, but may be containable to only that process, so that other processes in the system are unaffected.

Machine check exception handlers that attempt to recover must be thorough in their analysis and the corrective actions they take. The following guidelines should be used when writing such a handler:

- All status registers in the error-reporting banks must be examined to identify the cause of the machine check exception. Read MSR0000_0179 [Global Machine Check Capabilities (MCG_CAP)][Count] to determine the number of status registers visible to each core. The status registers are numbered from 0 to one less than the value found in MSR0000_0179[Count]. For example, if the Count field indicates five status registers are supported, they are numbered MC0_STATUS to MC4_STATUS.
- Check the valid bit in each status register (MC*i*_STATUS[Val]). The remainder of the MC*i*_STATUS register does not need to be examined when its valid bit is clear.
- When identifying the error condition, portable exception handlers should examine MC*i*_STATUS[Error Code] and [ErrorCodeExt].
- When logging errors, particularly those that are not recoverable, check MSR0000_017A [Global Machine Check Status (MCG_STAT)][EIPV] to see if the instruction pointer address pushed onto the exception handler stack is related to the machine check. If EIPV is clear, the address may not be related to the error.

- Check the valid MC*i*_STATUS registers to see if error recovery is possible. Error recovery is not possible when:
  - The processor context corrupt indicator (MC*i*_STATUS[PCC]) is set to 1.
  - The error overflow status indicator (MC*i*_STATUS[Over]) is set to 1. This indicates that more than one machine check error has occurred, but only one error is reported by the status register.

  If error recovery is not possible, the handler should log the error information and return to the operating system.
- Check MC*i*_STATUS[UC] to see if the processor corrected the error. If UC is set, the processor did not correct the error, and the exception handler must correct the error prior to attempting to restart the interrupted program. If the handler cannot correct the error, it should log the error information and return to the operating system.
- If MSR0000_017A [Global Machine Check Status (MCG_STAT)][RIPV] is set, the interrupted program can be restarted reliably at the instruction pointer address pushed onto the exception handler stack. If RIPV is clear, the interrupted program cannot be restarted reliably, although it may be possible to restart it for debugging purposes.
- Prior to exiting the machine check handler, be sure to clear MSR0000_017A [Global Machine Check Status (MCG_STAT)][MCIP]. MCIP indicates that a machine check exception is in progress. If this bit is set when another machine check exception occurs, the processor enters the shutdown state.
- When an exception handler is able to successfully log an error condition, clear the MC*i*_STATUS registers prior to exiting the machine check handler. Software is responsible for clearing at least MC*i*_STATUS[Val].

Additional machine check handler portability can be added by having the handler use the CPUID instruction to identify the processor and its capabilities. Implementation specific software can be added to the machine check exception handler based on the processor information reported by CPUID.

2.16.2.4.1 MCA Differentiation Between System-Fatal and Process-Fatal Errors

The bits MC*i*_STATUS[PCC], MSR0000_017A[RIPV], and MSR0000_017A[EIPV] form a hierarchy, used by software to determine the degree of corruption and recoverability in the system. Table 46 shows how these bits are interpreted.

Table 46: Error Scope Hierarchy

PCC	RIPV	EIPV	Comments
1	0	0	Error has corrupted system state (PCC=1), and the process program cannot be restarted (RIPV=0). The error is fatal to the system.
0	1	0/1	Error is confined to the process (PCC=0), and the process program can be restarted (RIPV=1).
	0	0/1	Corruption is confined to the process (PCC=0), but the exception is imprecise (RIPV=0) and the process program cannot be restarted. Continued operation of this process may not be possible without intervention, however system processing or other processes can continue with appropriate software clean up.

2.17 Sideband Interface (SBI)

The sideband interface (SBI) is an SMBus v2.0 compatible 2-wire processor slave interface. SBI is also referred as the Advanced Platform Management Link. All I2C v2.1 speeds are supported.

SBI is used to communicate with the Temperature Sensor Interface (SB-TSI).

2.17.1 SBI Processor Information

Processor access to the SBI configuration is via D18F3x1E4 [SBI Control]. The processor can access SB-TSI registers through D18F3x1E8 [SBI Address] and D18F3x1EC [SBI Data].

3 Registers

This section provides detailed field definitions for the register sets in the processor.

3.1 Register Descriptions and Mnemonics

Each register in this document is referenced with a mnemonic. Each mnemonic is a concatenation of the register-space indicator and the offset of the register. Here are the mnemonics for the various register spaces:

- **APICXX0**: APIC memory-mapped registers; XX0 is the hexadecimal byte address offset from the base address. The base address for this space is specified by MSR0000_001B [APIC Base Address (APIC_BAR)]. Unless otherwise specified, there is one set of these registers per core; each core may only access its own set of these registers. See 3.18 [APIC Registers].
- **CPUID FnXXXX_XXXX**: processor capabilities information returned by the CPUID instruction. See 3.19 [CPUID Instruction Registers]. Unless otherwise specified, there is one set of these registers per core; each core may only access its own set of these registers. See 3.19 [CPUID Instruction Registers].
- **DZFYxXXX**: PCI-defined configuration space; XXX specifies the hexadecimal byte address of the configuration register (this may be 2 or 3 digits); Y specifies the function number; Z specifies the hexadecimal device number; e.g., D18F3x40 specifies the register of device 18h, function 3, address 40h. See 2.7 [Configuration Space], for details about configuration space. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.3 [Device 0 Function 0 (Root Complex) Configuration Registers] through 3.14 [Device 18h Function 7 Configuration Registers].
- **FCRxXXXX_XXXX**: Fixed configuration registers used for various aspects of processor initialization; XXXX_XXXX is the hexadecimal address. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.16 [Fixed Configuration Space (FCR)].
- **GMMxXXXXX**: GPU memory mapped registers; XXXXX specifies the hexadecimal byte address offset (this may be 2 to 5 digits) from the base address register; The base address for this space is specified by D1F0x18 [Graphics Memory Mapped Registers Base Address]. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.17 [GPU Memory Mapped Registers].
- **IOXXX**: x86-defined input and output address space registers; XXX specifies the hexadecimal byte address of the IO instruction. This space includes IO-space configuration access registers IOCF8 [IO-Space Configuration Address] and IOCFC [IO-Space Configuration Data Port]. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.2 [IO Space Registers].
- **MSRXXXX_XXXX**: model-specific registers; XXXX_XXXX is the hexadecimal MSR number. This space is accessed through x86-defined RDMSR and WRMSR instructions. Unless otherwise specified, there is one set of these registers per core; each core may only access its own set of these registers. See 3.20 [MSRs - MSR0000_xxxx] through 3.23 [MSRs - MSRC001_1xxx].
- **PMCxXXX**: performance monitor events; XXX is the hexadecimal event counter number programmed into MSRC001_00[03:00] [Performance Event Select (PERF_CTL[3:0])][EventSelect]. Unless otherwise specified, there is one set of these registers per core; each core may only access its own set of these registers. See 3.24 [Performance Counter Events].
- **SMUxXX**: Internal SMU registers; XX specifies the hexadecimal byte address. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.15 [Internal System Management Unit (SMU) Registers].

Some register spaces contain an index/data register pair to provide access to an indexed register space. The register mnemonic for an indexed register is formed by concatenating the data register mnemonic with the index used to access the register. For example, an indexed register accessed through the index/data register pair D0F0xE0 and D0F0xE4 using the index 0130_8011h is given the register mnemonic D0F0xE4_x0130_8011.

Each mnemonic may specify the location of one or more registers that share the same base definition. A mnemonic that specifies more than one register will contain one or more ranges within braces. The ranges are specified as follows:

- Comma separated lists [A, B]: Define specific instances of a register, e.g., D0F3x[1,0]40 defines two registers D0F3x40 and D0F3x140.
- Colon separated ranges [A:B]: Defines all registers that contain the range between A and B. Examples:
  - D0F3x[50:40] defines five registers: D0F3x40, D0F3x44, D0F3x48, D0F3x4C, and D0F3x50.
  - D[8:4]F0x40 defines five registers: D4F0x40, D5F0x40, D6F0x40, D7F0x40, and D8F0x40.
  - D0F0xE4_x013[2:0]_0000 defines three registers: D0F0xE4_x0130_0000, D0F0xE4_x0131_0000, and D0F0xE4_x0132_0000.
- Colon separated ranges with a explicit step [A:B:stepC]: Defines the registers from B to A, where C specifies the offset between registers. For example, D0F3x[50:40:step8] defines three registers: D0F3x40, D0F3x48, and D0F3x50.
- If no step is specified, there is a default offset between registers which varies by register space:

Table 47: Default offset between registers by register space

Register space	Default offset
CPUID FnXXXX_XXXX	1
MSRXXXX_XXXX	1
APICXX0	16
Indexed registers	1
All others	4

- If a colon separated range defines a set of registers, and a subset of those registers are listed in tables within the register description, registers not listed are reserved unless defined elsewhere.

The processor includes a single set of IO-space and configuration-space registers. However, APIC, CPUID, and MSR register spaces are implemented once per core. Access to IO-space and configuration space registers may require software-level techniques to ensure that no more than one core attempts to access a register at a time.

The following is terminology found in the register descriptions.

Table 48: Terminology in register descriptions

Terminology	Description

BIOS	The recommended value to be set by AMD BIOS software. The format is defined to be BIOS: <integer-expression>.
	• If "BIOS:" occurs in a register field:
	  • The recommended value is applied to the field.
	• If "BIOS:" occurs after a register name but outside of a register field table row:
	  • The recommended value is applied to the width of the register.
	• E.g.: BIOS: 4h.
	• E.g.: BIOS: 1h if (F0x84[IsocEn]==1 && F0x68[DispRefModeEn]==0 && MSRC001_001F[EnConvertToNonIsoc]==0).

SBIOS	The recommended value to be set by system BIOS. See: BIOS.
See	There are multiple semantics for the see keyword, each differentiated by the content in which it appears. • Syntax1: See: register-mnemonic. Implies full width of register. • Syntax2: See: register-mnemonic[register-field-name]. • When the see statement is the first word of a register field then the current register field inherits the full definition of the referenced register field, including the field name. Any other definition overrides the inherited definition. • When the see statement is NOT the first word of a register field then the current register field inherits the full definition of the referenced register field, except the field name. Any other definition overrides the inherited definition.
Alias	The alias keyword allows the definition of a soft link between two registers. • For X is an alias of Y, X is a soft link to the register Y. • For X1, X2 are an alias of Y, both X1 and X2 are soft links to Y.

IF
THEN
ELSEIF
ELSE
ENDIF
	Allows conditional definition as a function of register fields. The syntax is:
	• IF (conditional-expression) THEN definition ENDIF.
	• IF (conditional-expression) THEN definition ELSE definition ENDIF.
	• IF (conditional-expression) THEN definition ELSEIF (conditional-expression) THEN definition ELSE definition ENDIF.

Access types	
Read	Capable of being read by software.
Read-only	Capable of being read but not written by software.
Write	Capable of being written by software.
Write-only	Write-only. Capable of being written by software. Reads are undefined.
Read-write	Capable of being written by software and read by software.
Set-by-hardware **Cleared-by-hardware** **Updated-by-hardware**	Register field is set high by hardware, cleared low by hardware, or updated by hardware.
Set-when-done **Cleared-when-done**	Register field is set high or cleared low by hardware when the operation is complete.
Write-once	After RESET_L is asserted, these registers may be written to once. After being written, they become read-only until the next RESET_L assertion. The write-once control is byte based. For example, software may write each byte of a write-once doubleword as four individual transactions. As each byte is written, that byte becomes read-only.

Table 48: Terminology in register descriptions

Terminology	Description
Write-1-to-clear	Software must write a 1 to the bit in order to clear it. Writing a 0 to these bits has no effect.
Write-1-only	Software can set the bit high by writing a 1 to it. Writes of 0 have no effect.
Reset-applied	Takes effect on warm reset.
GP-read	GP exception occurs on read; GP exception occurs on write; GP exception occurs on a read or a write.
GP-write	
GP-read-write	
Strap	Requires a strap configuration procedure to update. See 2.11.3.2.1 [Straps].
Per-core	One instance per core. Writes of these bits from one core only affect that core's register. Reads return the values appropriate to that core.
Per-node	One instance per node. See 3.1.1 [Northbridge MSRs In Multi-Core Products].
Field definitions	
Reserved	Field is reserved for future use. Software is required to preserve the state read from these bits when writing to the register. Software may not depend on the state of reserved fields nor on the ability of such fields to return the state previously written.
Unused	Field is reserved for future use. Software is not required to preserve the state read from these bits when writing to the register. Software may not depend on the state of unused fields nor on the ability of such fields to return the state previously written.
MBZ	Must be zero. If software attempts to set an MBZ bit to 1, a general-protection exception (#GP) occurs.
RAZ	Read as zero. Writes are ignored, unless RAZ is combined with write, write-1-only, or write-once.
Reset definitions	
Reset	The reset value of each register is provided below the mnemonic or in the field description. Unless otherwise noted, the register state matches the reset value when RESET_L is asserted (either a cold or a warm reset). An X in the reset value indicates that the register or field resets (warm or cold) to an unspecified state.
Cold reset	The field or register state is not affected by a warm reset (even if the field or register is labeled "Cold reset: X"); it is placed into the reset state when PWROK is de-asserted. See "Reset" above for the definition of "X" in a cold reset value.
Value	The current value of a read-only field or register. A value statement explicitly defines the value returned under all conditions including after reset events, and also defines the field or register as read-only. A field or register labeled "Value:" will not have a separate reset definition.

3.1.1 Northbridge MSRs In Multi-Core Products

MSRs that control Northbridge functions are shared between all cores in a multi-core processor (e.g. MSR0000_0410). If control of Northbridge functions is shared between software on all cores, software must ensure that only one core at a time is allowed to access the shared MSR.

3.1.2 Mapping Tables

The following mapping table types are defined.

3.1.2.1 Register Mapping

The register mapping table specifies the specific function for each register in a range of registers.

Table 100, for example, specifies that the GMMx281C register is for CS0, resulting in the register name "GMC DCT CS Base Address - CS0".

3.1.2.2 Index Mapping

The index mapping table is similar to the register mapping table, but specifies the register by index instead of by full register mnemonic.

Table 86, for example, specifies that the D18F2x9C_x0D0F_2002h register is for Clock 0 Pad Group 0, resulting in the register name "Clock Transmit Pre-Driver Calibration - Clock 0 Pad Group 0".

3.1.2.3 Field Mapping

The field mapping table maps the fields of a range of registers. The rows are the registers that are mapped. Each column specifies a field bit range that is mapped by that column for all rows. The cell at the intersection of the register row and the field bit range column specifies the suffix that is appended to the register field with a "_". "Reserved" specifies that the field is reserved for the register of that row.

Table 54, for example, specifies that the field at D18F2x9C_x0000_0[1:0]01 bits 31:24 should have the suffix "Byte 3" resulting in WrDatGrossDly_Byte3 and WrDatFineDly_Byte3 (field names do not contain spaces).

3.1.2.4 Broadcast Mapping

The broadcast mapping table maps a register address to a range of register addresses. The register address is formed by the concatenation of the row address with the column address. The cell at the intersection of the row and column address is a range of register addresses that will be read or written as a group when the row and column address is read or written.

Table 69, for example, specifies that a read or write to D18F2x98[31:0]==0D0F_0F02h will result in a broadcast read or write to the D18F2x9C_x0D0F_0[7:0]02 range of registers.

3.1.2.5 Reset Mapping

The reset mapping table specifies the reset, cold reset, or value for each register in a range of registers.

Table 102, for example, specifies that the CPUID Fn0000_0000_EBX register has a value of 6874_7541h, with a comment of "The ASCII characters "h t u A"".

3.1.2.6 Valid Values

The valid values table defines the valid values for one or more register fields. The valid values table is equivalent in function to the Bits/Definition tables in register fields (e.g. D18F3x44[WDTBaseSel]) and is most often used when the table becomes too large and unwieldy to be included into the register field (e.g. Table 70 [Valid values for D18F2x9C_x0D0F_0[F,7:0]02[TxPreP, TxPreN]]).

3.1.2.7 BIOS Recommendations

The BIOS recommendations table defines "BIOS:" recommendations that are conditional and complex enough

to warrant a table. All cells under the "Condition" header for a given row are ANDed to form the condition for the values to the right of the condition.

Table 8 [Recommended buffer settings], for example, specifies the BIOS recommendations for some fields of registers D18F3x6C, D18F3x74, D18F3x7C, and D18F3x17C. The equivalent BIOS recommendation for D18F3x6C[UpLoPreqDBC] is:

IF (D0F0x98_x1E[HiPriEn]==0) THEN BIOS: Eh.
ELSE BIOS: Dh.
ENDIF.

3.2 IO Space Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention.

IOCF8 IO-Space Configuration Address

Reset: 0000_0000h.

IOCF8 [IO-Space Configuration Address], and IOCFC [IO-Space Configuration Data Port], are used to access system configuration space, as defined by the PCI specification. IOCF8 provides the address register and IOCFC provides the data port. Software sets up the configuration address by writing to IOCF8. Then, when an access is made to IOCFC, the processor generates the corresponding configuration access to the address specified in IOCF8. See 2.7 [Configuration Space].

IOCF8 may only be accessed through aligned, doubleword IO reads and writes; otherwise, the accesses are passed to the appropriate link. Accesses to IOCF8 and IOCFC received from a link are treated as all other IO transactions received from a link and are forwarded based on the settings in D18F1xC0 [IO-Space Base]. IOCF8 and IOCFC in the processor are not accessible from a link.

Bits	Description
31	**ConfigEn: configuration space enable**. Read-write. 1=IO read and write accesses to IOCFC are translated into configuration cycles at the configuration address specified by this register. 0=IO read and write accesses to IOCFC are passed to the appropriate link and no configuration access is generated.
30:28	Reserved.
27:24	**ExtRegNo: extended register number**. Read-write. ExtRegNo provides bits[11:8] and RegNo provides bits[7:2] of the byte address of the configuration register. ExtRegNo is reserved unless it is enabled by MSRC001_001F[EnableCf8ExtCfg].
23:16	**BusNo: bus number**. Read-write. Specifies the bus number of the configuration cycle.
15:11	**Device: device number**. Read-write. Specifies the device number of the configuration cycle.
10:8	**Function**. Read-write. Specifies the function number of the configuration cycle.
7:2	**RegNo: register address**. Read-write. See IOCF8[ExtRegNo].
1:0	Reserved.

IOCFC IO-Space Configuration Data Port

Reset: 0000_0000h.

Bits	Description
31:0	**ConfigData**. Read-write. See IOCF8 for details about this port.

3.3 Device 0 Function 0 (Root Complex) Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space.

D0F0x00 Device/Vendor ID

Reset: 1510_1022h.

Bits	Description
31:16	**DeviceID: device ID**. Read-only.
15:0	**VendorID: vendor ID**. Read-only.

D0F0x04 Status/Command

Reset: 0220_0004h.

Bits	Description
31	**ParityErrorDetected: parity error detected**. Read-only.
30	**SignaledSystemError: signaled system error**. Read; write-1-to-clear. 1=FCH generated a system error.
29	**ReceivedMasterAbort: received master abort**. Read; write-1-to-clear.
28	**ReceivedTargetAbort: received target abort**. Read; write-1-to-clear.
27	**SignalTargetAbort: signaled target abort**. Read-only.
26:25	**DevselTiming: DEVSEL# Timing**. Read-only.
24	Reserved.
23	**FastBackCapable: fast back-to-back capable**. Read-only.
22	Reserved.
21	**PCI66En: 66 MHz capable**. Read-only.
20	**CapList: capability list**. Read-only. 1=Capability list supported.
19:10	Reserved.
9	**FastB2BEn: fast back-to-back enable**. Read-write.
8	**SerrEn: system error enable**. Read-write.
7	Reserved.
6	**ParityErrorEn: parity error response enable**. Read-only.
5	**PalSnoopEn: VGA palette snoop enable**. Read-only.
4	**MemWriteInvalidateEn: memory write and invalidate enable**. Read-only.
3	**SpecialCycleEn: special cycle enable**. Read-only.

2	**BusMasterEn: bus master enable**. Read-only.
1	**MemAccessEn: memory access enable**. Read-write. This bit controls if memory accesses by this device are accepted or not. 1=Enabled. 0=Disabled.
0	**IoAccessEn: IO access enable**. Read-only.

D0F0x08 Class Code/Revision ID

Reset: 0600_00xxh.

Bits	Description
31:8	**ClassCode: class code**. Read-only. Provides the host bridge class code as defined in the PCI specification.
7:0	**RevID: revision ID**. Read-only.

D0F0x0C Header Type

Reset: 0000_0000h.

Bits	Description
31:24	**BIST**. Read-only.
23:16	**HeaderTypeReg**. Read-only. 00h=Single function device.
15:8	**LatencyTimer**. Read-write.
7:0	**CacheLineSize**. Read-only.

D0F0x2C Subsystem and Subvendor ID

Reset: 1510_1022h.

Bits	Description
31:16	**SubsystemID**. Read-only.
15:0	**SubsystemVendorID**. Read-only.

D0F0x34 Capabilities Pointer

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**CapPtr: capabilities pointer**. Read-only. There is no capability list.

D0F0x4C PCI Control

Reset: 0000_2002h.

Bits	Description
31:27	Reserved.
26	**HPDis: hot plug message disable**. Read-write. 1=Hot plug message generation is disabled.
25:24	Reserved.
23	**MMIOEnable: memory mapped IO enable**. Read-write. 1=Decoding of MMIO cycles is enabled.
22:15	Reserved.
14:12	**CfgRdTime**. Read-write. BIOS: 010b. Specifies the propagation delay for read data on the configuration bus. Bits Definition 111b-000b <7-CfgRdTime> clocks
11	**CRS**: **configuration request retry detected**. Read; set-by-hardware; write-1-to-clear. 1=Configuration request retry was detected.
10:6	Reserved.
5	**SerrDis**: s**ystem error message disable**. Read-write. 1=The generation of SERR messages is disabled.
4	**PMEDis: PME disable**. Read-write. 1=The generation of PME messages is disabled.
3	**Cf8Dis: CF8 disable**. Read-write. 1=Configuration accesses through IO address CF8h to this device are disabled.
2	Reserved.
1	**ApicEnable: APIC enable**. Read-write. 1=APIC is enabled.
0	**Function1Enable: device 0 function 1 enable**. Read-write. 1=Configuration accesses to device 0 function 1 are enabled.

D0F0x60 Miscellaneous Index

Reset: 0000_0000h.

The index/data pair registers D0F0x60 and D0F0x64 is used to access the registers D0F0x64_x[FF:00]. To read or write to one of these register, the address is written first into the address register D0F0x60 and then the data are read or written by read or write the data register D0F0x64.

Bits	Description
31:8	Reserved.
7	**MiscIndWrEn: miscellaneous index write enable**. Read-write. If set writes to D0F0x64 are enabled.
6:0	**MiscIndAddr: miscellaneous index register address**. Read-write.

D0F0x64 Miscellaneous Index Data

Reset: 0000_0042h. See D0F0x60.

Bits	Description
31:0	**MiscIndData: miscellaneous index data register**. Read-write.

D0F0x64_x00 Northbridge Control

Reset: 0000_0002h.

Bits	Description
31:8	Reserved.
7	**HwInitWrLock**. Read-write. 1=Lock HWInit registers. 0=Unlock HWInit registers.
6	**NbFchCfgEn: device 8 enable**. Read-write. 1=Bridge device 8 is enabled.
5:0	Reserved.

D0F0x64_x0B IOC Link Control

Reset: 0000_0000h.

Bits	Description
31:24	Reserved.
23	**IocFchSetPmeTurnOffEn**. Read-write. 1=Enables the PME_Turn_Off/PME_To_Ack mechanism between northbridge and FCH.
22	Reserved.
21	**IocFchSetPowEn: set slot power enable**. Read-write. 1=Enables sending set_slot_power_limit/scale messages to the FCH.
20	**SetPowEn: set slot power enable**. Read-write. 1=Enables sending set_slot_power messages to the FCH.
19:0	Reserved.

D0F0x64_x0C IOC Bridge Control

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7	**Dev7BridgeDis**. Read-write. 1=Bus 0, device 7 bridge functionality is hidden.
6	**Dev6BridgeDis**. Read-write. 1=Bus 0, device 6 bridge functionality is hidden.
5	**Dev5BridgeDis**. Read-write. 1=Bus 0, device 5 bridge functionality is hidden.
4	**Dev4BridgeDis**. Read-write. 1=Bus 0, device 4 bridge functionality is hidden.
3:0	Reserved.

D0F0x64_x16 IOC Advanced Error Reporting Control

Reset: 0000_0001h.

Bits	Description
31:1	Reserved.
0	**AerUrMsgEn: AER unsupported request message enable**. Read-write. 1=AER unsupported request messages are enabled.

D0F0x64_x19 Top of Memory 2 Low

Reset: 0000_0000h.

Bits	Description
31:23	**Tom2[31:23]: top of memory 2**. Read-write. BIOS: MSRC001_001D[Tom2[31:23]]. This field specifies the maximum system address for upstream read and write transactions that are forwarded to the host bridge. All addresses less than or equal to this system address are forwarded to DRAM and are not checked to determine if the transaction is a peer-to-peer transaction. All upstream reads with addresses greater than this system address are master aborted.
22:1	Reserved.
0	**TomEn: top of memory enable**. Read-write. BIOS: MSRC001_0010[MtrrTom2En]. 1=Top of memory check enabled.

D0F0x64_x1A Top of Memory 2 High

Reset: 0000_0000h.

Bits	Description
31:4	Reserved.
3:0	**Tom2[35:32]: top of memory 2**. Read-write. BIOS: MSRC001_001D[Tom2[35:32]]. See D0F0x64_x19[Tom2].

D0F0x64_x1C Internal Graphics PCI Control 1

Reset: 0000_0000h.

Bits	Description
31:24	Reserved.
23	**RcieEn**. IF (D0F0x64_x1C[WriteDis]==1) THEN Read-only. ELSE Read-write. ENDIF. BIOS: 1. 1=Root complex integrated endpoint mode. 0=Legacy PCI device mode.
22:18	Reserved.
17	**F0En**. IF (D0F0x64_x1C[WriteDis]==1) THEN Read-only. ELSE Read-write. ENDIF. BIOS: 1. 1=Internal graphics enabled. 0=Internal Graphics disabled.
16	**IoBarDis**. IF (D0F0x64_x1C[WriteDis]==1) THEN Read-only. ELSE Read-write. ENDIF. 1=Graphics IO base address register disabled. 0= Graphics IO base address register enabled.
15:12	Reserved.

11	**Audio64BarEn**. IF (D0F0x64_x1C[WriteDis]==1) THEN Read-only. ELSE Read-write. ENDIF. Controls the size of the audio BAR. 1=64-bit BAR. 0=32-bit BAR.
10	**AudioNonlegacyDeviceTypeEn**. IF (D0F0x64_x1C[WriteDis]==1) THEN Read-only. ELSE Read-write. ENDIF. BIOS: 0. 1=PCIe device. 0=Legacy PCI device.
9	**MsiDis**. IF (D0F0x64_x1C[WriteDis]==1) THEN Read-only. ELSE Read-write. ENDIF. BIOS: 0. 1=MSI interrupts disabled. 0=MSI interrupts enabled.
8	**AudioEn**. IF (D0F0x64_x1C[WriteDis]==1) THEN Read-only. ELSE Read-write. ENDIF. BIOS: 1. 1=Enable HD audio over HDMI™ or DisplayPort.
7	Reserved.
6	**RegApSize**. IF (D0F0x64_x1C[WriteDis]==1) THEN Read-only. ELSE Read-write. ENDIF. BIOS: 1. Specifies the size of the graphics register aperture. 0=64KB. 1=256KB.
5:3	**MemApSize**. IF (D0F0x64_x1C[WriteDis]==1) THEN Read-only. ELSE Read-write. ENDIF. Specifies the size of the frame buffer aperture. Bits Definition Bits Definition 000b 128MB 100b 512MB 001b 256MB 101b 1GB 010b 64MB 110b 2GB 011b 32MB 111b 4GB
2	**F064BarEn**. IF (D0F0x64_x1C[WriteDis]==1) THEN Read-only. ELSE Read-write. ENDIF. 1=64-bit base address registers. 0=32-bit base address registers.
1	**F0NonlegacyDeviceTypeEn**. IF (D0F0x64_x1C[WriteDis]==1) THEN Read-only. ELSE Read-write. ENDIF. BIOS: 0. 1=PCIe device. 0=Legacy PCI device.
0	**WriteDis**. IF (D0F0x64_x1C[WriteDis]==1) THEN Read-only. ELSE Read-write. ENDIF. 1=Register is read-only and the GPU PCI interface is configured. 0=Register is read-write. See 2.15.1 [GPU PCI Interface].

D0F0x64_x1D Internal Graphics PCI Control 2

Reset: 0000_0000h.

Bits	Description
31:4	Reserved.
3	**Vga16En: VGA IO 16 bit decoding enable**. Read-write. 1=Address bits 15:10 for VGA IO cycles are decoded. 0=Address bits 15:10 for VGA IO cycles are ignored.
2	Reserved.
1	**VgaEn: VGA enable**. Read-write. Affects the response by the internal graphics to compatible VGA addresses when IntGfxAsPcieEn=1. 1=The internal graphics decodes the following accesses: • Memory accesses in the range of A0000h to BFFFFh. • IO address where address bits 9:0 are in the ranges of 3B0h to 3BBh or 3C0h to 3DFh. For IO cycles the decoding of address bits 15:10 depends on Vga16En.
0	**IntGfxAsPcieEn: internal graphics is a root complex integrated device**. Read-write. BIOS: 1. 1=Integrated graphics is device 1 on bus 0 and operates as a root complex integrated device. Software must program D1F0x04[BusMasterEn]=1 before programming IntGfxAsPcieEn=1. 0=Integrated graphics is located behind a PCI-to-PCI bridge and is device 5 on the bus behind the bridge. The bridge device is device 1 on bus 0.

D0F0x64_x46 IOC Features Control

Reset: 0000_3063h.

Bits	Description
31:17	Reserved.
16	**Msi64bitEn: 64-bit MSI enable**. Read-write. 1=64-bit MSI support enabled. 0=64-bit MSI support disabled.
15:3	Reserved.
2:1	**P2PMode: peer-to-peer mode**. Read-write. BIOS: 00b. Specifies how upstream write transactions above D0F0x64_x19[Tom2] are completed. Bits: Definition 00b: Master abort writes that do not hit one of the internal PCI bridges. Forward writes that hit one of the internal PCI bridges to the bridge. 11b-01b: Reserved.
0	Reserved.

D0F0x64_x4D SMU Request Port

Reset: 0000_0000h. See 3.15 [Internal System Management Unit (SMU) Registers].

Bits	Description
31:26	Reserved.
25	**ReqType**. Read-write. 1=Write. 0=Read.
24	**ReqToggle: request toggle**. Read-write.
23:16	**SmuAddr**. Read-write. Specifies the SMU register address.
15:0	**WriteData**. Read-write. Specifies the data written to the SMU.

D0F0x64_x4E SMU Read Data

Reset: 0000_0000h. See 3.15 [Internal System Management Unit (SMU) Registers].

Bits	Description
31:0	**SmuReadData**. Read-only; updated-by-hardware. Returns the data read from the SMU.

D0F0x64_x5[B,9,7,5] IOC PCIe® Device Control

Reset: 0000_0000h.

Bits	Description
31:21	Reserved.

20	**SetPowEn: set slot power enable**. Read-write. IF ((REG==D0F0x64_x55) && (D0F0x64_x0C[Dev4BridgeDis]==0)) THEN BIOS: 1. ELSEIF ((REG==D0F0x64_x57) && (D0F0x64_x0C[Dev5BridgeDis]==0)) THEN BIOS: 1. ELSEIF ((REG==D0F0x64_x59) && (D0F0x64_x0C[Dev6BridgeDis]==0)) THEN BIOS: 1. ELSEIF ((REG==D0F0x64_x5B) && (D0F0x64_x0C[Dev7BridgeDis]==0)) THEN BIOS: 1. ELSE BIOS: 0. 1=Enables the set_slot_power message to the FCH. ENDIF.
19:0	Reserved.

D0F0x64_x6A Voltage Control

Reset: 0000_0000h. See 2.5.1.5.2 [Software-Initiated Voltage Transitions].

Bits	Description
31:5	Reserved.
4:3	**VoltageLevel**. Read-write. Specifies the VID requested when software toggles D0F0x64_x6A[VoltageChangeReq]. This field indexes into D18F3x15C as follows: Bits — VID code 00b — D18F3x15C[SclkVidLevel0] 01b — D18F3x15C[SclkVidLevel1] 10b — D18F3x15C[SclkVidLevel2] 11b — D18F3x15C[SclkVidLevel3]
2	**VoltageChangeReq**. Read-write. Software toggles this field to make VDDCR_NB voltage requests.
1	**VoltageChangeEn**. Read-write. Specifies whether changes to D0F0x64_x6A[VoltageChangeReq] causes voltage change requests. 1=Requests occur. 0=Requests do not occur.
0	**VoltageForceEn**. Read-write. If D0F0x64_x6A[VoltageChangeEn]==1, this field specifies whether changes to D0F0x64_x6A[VoltageChangeReq] cause forced voltage changes. 1=Voltage changes are forced. 0=Voltage changes are not forced.

D0F0x64_x6B Voltage Status

Cold reset: 0000_0006h. See 2.5.1.5.2 [Software-Initiated Voltage Transitions].

Bits	Description
31:3	Reserved.
2:1	**CurrentVoltageLevel**. Read-only; updated-by-hardware. Specifies the current voltage level requested by D0F0x64_x6A. See D0F0x64_x6A[VoltageLevel]. To determine the current voltage level, software must poll on this field until two consecutive reads return the same value.
0	**VoltageChangeAck**. Read-only; updated-by-hardware. Specifies whether the voltage change requested by D0F0x64_x6A[VoltageChangeReq] is complete.

D0F0x78 Scratch

Cold reset: 0000_0000h.

Bits	Description
31:0	**Scratch**. Read-write. This register does not control any hardware.

D0F0x7C IOC Configuration Control

Reset: 0000_0000h.

Bits	Description
31:1	Reserved.
0	**ForceIntGFXDisable: internal graphics disable**. Read-write. Setting this bit disables bridge device 1 on bus 0 and all devices behind this bridge.

D0F0x84 Link Arbitration

Reset: 0300_0000h.

Bits	Description
31:10	Reserved.
9	**PmeTurnOff: PME_Turn_Off message trigger**. Read-write. Reset: 0. 1=Trigger a PME_Turn_Off message to all downstream devices if PmeMode=1.
8	**PmeMode: PME message mode**. Read-write. Reset: 0. 1=PME_Turn_Off message is triggered by writing PmeTurnOff. 0=PME_Turn_Off message is triggered by a message from the FCH.
7:5	Reserved.
4	**Ev6Mode: EV6 mode**. Read-write. Reset: 0. BIOS: 1. 1=The links decode the memory range from 640K to 1M.
3:0	Reserved.

D0F0x90 Northbridge Top of Memory

Reset: 0000_0000h.

Bits	Description
31:23	**TopOfDram**. Read-write. BIOS: MSRC001_001A[TOM[31:23]]. Specifies the address that divides between MMIO and DRAM. From TopOfDram to 4G is MMIO; below TopOfDram is DRAM. See 2.4.3 [Access Type Determination].
22:0	Reserved.

D0F0x94 Northbridge ORB Configuration Offset

Reset: 0000_0000h.
The index/data pair D0F0x94 and D0F0x98 are used to access D0F0x98_x[FF:00]. To read or write to one of these register, the address is written first into the address register D0F0x94 and then the data are read or written by read or write the data register D0F0x98.

Bits	Description
31:9	Reserved.
8	**OrbIndWrEn: ORB index write enable**. Read-write. 1=Writes to D0F0x98 are enabled.
7	Reserved.
6:0	**OrbIndAddr: ORB index register address**. Read-write.

D0F0x98 Northbridge ORB Configuration Data Port

Reset: 0000_0000h. See D0F0x94.

Bits	Description
31:0	**OrbIndData: ORB index data register**. Read-write.

D0F0x98_x06 ORB Downstream Control 0

Reset: 0000_0000h.

Bits	Description
31:27	Reserved.
26	**UmiNpMemWrEn**. Read-write. BIOS: See 2.11.3.1. 1=NP protocol over UMI for memory-mapped writes targeting LPC enabled. This bit may be set to avoid a deadlock condition.
25:0	Reserved.

D0F0x98_x07 ORB Upstream Arbitration Control 0

Reset: 0000_0000h.

Bits	Description
31:16	Reserved.
15	**DropZeroMaskWrEn**. Read-write. BIOS: 1. 1=Drop byte write request that have all bytes masked. 0=Forward byte write request that have all bytes masked.
14	**MSIHTIntConversionEn**. Read-write. BIOS: 1. 1=MSI to HT interrupt conversion enabled.
13:1	Reserved.
0	**IocBwOptEn**. Read-write. BIOS: 1. 1=Enable optimization of byte writes by detecting consecutive doubleword masks and translating the request to doubleword writes.

D0F0x98_x08 ORB Upstream Arbitration Control 1

Reset: 0008_0008h. BIOS: 0001_0008h.

This register specifies the weights of the weighted round-robin arbiter in stage 1 of the upstream arbitration for non-posted reads.

Bits	Description
31:24	Reserved.
23:16	**NpWrrLenC**. Read-write. This field defines the maximum number of non-posted read requests from the GPU that are serviced before the arbiter switches to the next client.
15:8	Reserved.
7:0	**NpWrrLenA**. Read-write. This field defines the maximum number of non-posted read requests from IOC that are serviced before the arbiter switches to the next client.

D0F0x98_x09 ORB Upstream Arbitration Control 2

Reset: 0800_0008h. BIOS: 0100_0008h.

This register specifies the weights of the weighted round-robin arbiter in stage 1 of the upstream arbitration for posted writes.

Bits	Description
31:24	**PWrrLenD**. Read-write. This field defines the maximum number of posted write requests from the GPU that are serviced before the arbiter switches to the next client.
23:8	Reserved.
7:0	**PWrrLenA**. Read-write. This field defines the maximum number of posted write requests from the IOC that are serviced before the arbiter switches to the next client.

D0F0x98_x0C ORB Upstream Arbitration Control 5

Reset: 0000_0808h.

This register specifies the weights of the weighted round-robin arbiter in stage 2 of the upstream arbitration.

Bits	Description
31	Reserved.
30	**StrictSelWinnerEn**. Read-write. BIOS: 1. 1=Select arbitration winner when TX is idle and the FIFO is not full. 0=Select arbitration winner when TX is idle.
29:16	Reserved.
15:8	**GcmWrrLenB**. Read-write. BIOS: 08h. This field defines the maximum number of posted write requests from stage 1 that are getting serviced in the round-robin before the stage 2 arbiter switches to the next client.
7:0	**GcmWrrLenA**. Read-write. BIOS: 08h. This field defines the maximum number of non-posted read requests from stage 1 that are getting serviced in the round-robin before the stage 2 arbiter switches to the next client.

D0F0x98_x0E ORB MSI Interrupt Remap

Reset: 0000_FF0Ch.

Bits	Description
31:24	Reserved.
23:16	**MsiHtRsvIntVector**. Read-write. BIOS: 00h. This field defines the interrupt vector used when an MSI interrupt is received that has a reserved delivery mode field if MsiHtRsvIntRemapEn==1.
15:8	**MsiHtRsvIntDestination**. Read-write. BIOS: FFh. This field defines the interrupt destination used when an MSI interrupt is received that has a reserved delivery mode field when MsiHtRsvIntRemapEn==1.
7	Reserved.
6	**MsiHtRsvIntDM**. Read-write. BIOS: 0. Defines the interrupt destination mode when an MSI interrupt is received that has a reserved delivery mode field if MsiHtRsvIntRemapEn==1.

5	**MsiHtRsvIntRqEoi**. Read-write. BIOS: 0. Specifies the REQEIOI state when an MSI interrupt is received that has a reserved delivery mode field if MsiHtRsvIntRemapEn==1.
4:2	**MsiHtRsvIntMt**. Read-write. BIOS: 011b. Specifies the message type used when an MSI interrupt is received that has a reserved delivery mode field if MsiHtRsvIntRemapEn==1.
1	Reserved.
0	**MsiHtRsvIntRemapEn**. Read-write. BIOS: 1. 1=Remapping of MSI interrupts with reserved delivery mode to the interrupt programmed into this register enabled.

D0F0x98_x1E ORB Receive Control 0

Reset: 0000_0000h.

Bits	Description
31:2	Reserved.
1	**HiPriEn**. Read-write. BIOS: 0. 1=High priority channel enabled. See Table 8.
0	Reserved.

D0F0x98_x28 ORB Transmit Control 0

Reset: 0000_0000h.

Bits	Description
31:2	Reserved.
1	**ForceCoherentIntr**. Read-write. BIOS: 1. 1=Interrupt requests are forced to have coherent bit set.
0	**SmuPmInterfaceEn**. Read-write. BIOS: 1. 1=SMU to ORB power management interface enabled.

D0F0x98_x2C ORB Clock Control

Reset: 000F_0000h.

Bits	Description
31:16	**WakeHysteresis**. Read-write. BIOS: 64h. Specifies the amount of time hardware waits after ORB becomes idle before de-asserting the wake signal to the NB. Wait time = WakeHysteresis * 50 ns. Values less than 64h may result in undefined behavior.
15:2	Reserved.
1	**DynWakeEn**. Read-write. BIOS: 1. 1=Enable dynamic toggling of the wake signal between ORB and NB. 0=Disable dynamic toggling of the wake signal.
0	Reserved.

D0F0xE0 Link Index Address

Reset: 0130_8001h.

D0F0xE0 and D0F0xE4 are used to access D0F0xE4_x[FFFF_FFFF:0000_0000]. To read or write to one of these register, the address is written first into the address register D0F0xE0 and then the data is read from or written to the data register D0F0xE4.

Bits	Description
31:24	**BlockSelect: block select**. Read-write. This field is used to select the specific register block to access. Bits: Definition 00h: Reserved 01h: GPP link registers FFh-02h: Reserved
23:16	**FrameType: frame type**. Read-write. This field is used to select the type of register block to access. Bits: Definition 00h: Reserved 01h: Link core registers 07h-02h: Reserved 08h: Impedance controller registers 0Fh-09h: Reserved 10h: Phy interface block registers 1Fh-11h: Reserved 20h: Phy registers 2Fh-21h: Reserved 30h: Wrapper registers FFh-31h: Reserved
15:0	**PcieIndxAddr: index address**. Read-write.

D0F0xE4 Link Index Data

Reset: DEAD_BEEFh. See D0F0xE0.

Bits	Description
31:0	**PcieIndxData: index data**. Read-write.

3.3.1 IO Link Registers

D0F0xE4_x0101_0010 IO Link Control 1

Reset: 8063_0800h.

Bits	Description
31:13	Reserved.
12:10	**RxUmiAdjPayloadSize**. Read-write. BIOS: 100b. Payload size for the UMI. Bits: Definition: Bits: Definition 00xb: Reserved.: 100b: 64 bytes 010b: 16 bytes: 101b: Reserved. 011b: 32 bytes.: 11xb: Reserved.
9	**UmiNpMemWrite: memory write mapping enable**. Read-write. BIOS: See 2.11.3.1. 1=Internal non-posted memory writes are transferred to UMI.
8:1	Reserved.
0	**HwInitWrLock: hardware initialization write lock**. Read-write. 1=Lock HWInit registers. 0=Unlock HWInit registers.

D0F0xE4_x0101_001C IO Link Control 2

Reset: 0000_0000h.

Bits	Description
31:11	Reserved.
10:6	**TxArbMstLimit: transmitter arbitration master limit**. Read-write. BIOS: 4h. Defines together with TxArbSlvLimit a round robin arbitration pattern for downstream accesses. TxArbMstLimit defines the weight for downstream CPU requests and TxArbSlvLimit for the downstream read responses.
5:1	**TxArbSlvLimit: transmitter arbitration slave limit**. Read-write. BIOS: 4h. See TxArbMstLimit for details
0	**TxArbRoundRobinEn: transmitter round robin arbitration enabled**. Read-write. BIOS: 1. 1=Enable transmitter round robin arbitration. 0=Disable transmitter round robin arbitration.

D0F0xE4_x0101_0020 IO Link Chip Interface Control

Reset: 0000_0050h.

Bits	Description
31:10	Reserved.
9	**CiRcOrderingDis: chip interface root complex ordering disable**. Read-write. 0=Root complex ordering logic is enabled. 1=Root complex ordering logic is disabled.
8:0	Reserved.

D0F0xE4_x0101_0040 IO Link Phy Control

Reset: 0000_0000h.

Bits	Description
31:16	Reserved.
15:14	**PElecIdleMode: electrical idle mode for physical layer**. Read-write. BIOS: 10b. Defines which electrical idle signal is used, either inferred by link controller or from the phy. Bits — Definition 00b — Gen1 - entry=phy, exit=phy; Gen2 - entry=infer, exit=phy 01b — Gen1 - entry=infer, exit=phy; Gen2 - entry=infer, exit=phy 10b — Gen1 - entry=phy, exit=phy; Gen2 - entry=phy, exit=phy 11b — Reserved
13:0	Reserved.

D0F0xE4_x0101_00B0 IO Link Strap Control

Reset: 0000_0001h.

Bits	Description
31:3	Reserved.

2	**StrapF0MsiEn**. Read-write. BIOS: 1. 1=MSI support enabled. 0=MSI support disabled.
1:0	Reserved.

D0F0xE4_x0101_00C0 IO Link Strap Miscellaneous

Bits	Description
31:30	Reserved.
29	**StrapMstAdr64En**. Read-write. Reset: 0. 1=Enable 64 bit address support for MSI. 0=Disable 64 bit address support for MSI. D0F0xE4_x0101_00B0[StrapF0MsiEn] must be 1 before StrapMstAdr64En is programmed to 1.
28	**StrapReverseAll**. Read-write. Reset: 0.
27:0	Reserved.

D0F0xE4_x0101_00C1 IO Link Strap Miscellaneous

Bits	Description
31:2	Reserved.
1	**StrapGen2Compliance**. Read-write. Reset: 1.
0	**StrapLinkBwNotificationCapEn**. Read-write. Reset: 0.

3.3.2 PIF Registers

D0F0xE4_x0110_0011 PIF Pairing

Reset: 0200_0000h.

Table 49: Index addresses for D0F0xE4_x0110_0011

D0F0xE0[31:16]	D0F0xE0[15:0]
	0011h
0110h	GPPFCH PIF

Bits	Description
31:26	Reserved.
25	**MultiPif: x16 link**. Read-write. BIOS: 0.
24:17	Reserved.
16	**X8Lane70: x8 link lanes 7:0**. Read-write. BIOS: See Table 31. 1=Lanes 7:0 are paired to create a x8 link.
15:13	Reserved.
12	**X4Lane52: x4 link lanes 5:2**. Read-write. BIOS: See Table 31. 1=Lanes 5:2 are paired to create a x4 link.
11:10	Reserved.

9	**X4Lane74: x4 link lanes 7:4**. Read-write. BIOS: See Table 31. 1=Lanes 7:4 are paired to create a x4 link.
8	**X4Lane30: x4 link lanes 3:0**. Read-write. BIOS: See Table 31. 1=Lanes 3:0 are paired to create a x4 link.
7:4	Reserved.
3	**X2Lane76: x2 link lanes 7:6**. Read-write. BIOS: See Table 31. 1=Lanes 7:6 are paired to create a x2 link.
2	**X2Lane54: x2 link lanes 5:4**. Read-write. BIOS: See Table 31. 1=Lanes 5:4 are paired to create a x2 link.
1	**X2Lane32: x2 link lanes 3:2**. Read-write. BIOS: See Table 31. 1=Lanes 3:2 are paired to create a x2 link.
0	**X2Lane10: x2 link lanes 1:0**. Read-write. BIOS: See Table 31. 1=Lanes 1:0 are paired to create a x2 link.

3.3.3 Wrapper Registers

D0F0xE4_x0130_0000 BIF Core Feature Enable

Bits	Description
31:24	Reserved.
23	**StrapBifAriEn**. Read-write; strap. Reset: 0b. BIOS: 0. 1=Enable alternate requestor ID ECN support.
22:0	Reserved.

D0F0xE4_x0130_0002 Link Speed Control

Reset: 0000_0004h.

Bits	Description
31:3	Reserved.
2	**StrapPllCmpFreqMode**. Read-write; strap. 1=5 GHz. 0=2.5 GHz.
1:0	Reserved.

D0F0xE4_x0130_0080 Link Configuration

Bits	Description
31:4	Reserved.
3:0	**StrapBifLinkConfig**. Read-write; strap. Reset: Product-specific. BIOS: See Table 31. This field configures the GPP links.

Bits	Definition	Bits	Definition
0000b	Reserved	0011b	1 x2 IO Link, 2 x1 IO Links
0001b	x4 IO Link	0100b	4 x1 IO Links
0010b	2 x2 IO Links	1111b-0101b	Reserved

D0F0xE4_x0130_8002 Subsystem and Subvendor ID Control

Cold reset: 0000_0000h.

Bits	Description
31:16	**SubsystemID**. Read-write. Specifies the subsystem ID for PCIe devices 4 through 8. See D[8:4]F0xB4[SubsystemID].
15:0	**SubsystemVendorID**. Read-write. Specifies the subsystem vendor ID for PCIe devices 4 through 8. See D[8:4]F0xB4[SubsystemVendorID].

D0F0xE4_x0130_8011 Link Transmit Clock Gating Control

Bits	Description
31	**StrapBifValid**. Read-write. Reset: 0. 0=Straps are latched. 1=Straps are not latched.
30:17	Reserved.
16	**RcvrDetClkEnable**. Read-write. Reset: 0. 1=Enable receiver detection for hot-plug capability.
15:0	Reserved.

D0F0xE4_x0130_80F0 BIOS Timer

Reset: 0000_0000h.

Bits	Description
31:0	**MicroSeconds**. Read-write; updated-by hardware. This field increments once every microsecond when the timer is enabled. The counter will roll over and continue counting when it reaches its FFFF_FFFFh. A write to this register causes the counter to reset and begin counting from the value written.

D0F0xE4_x0130_80F1 BIOS Timer Control

Reset: 0000_0064h.

Bits	Description
31:8	Reserved.
7:0	**ClockRate**. Read-write. BIOS: 00h. Specifies the frequency of the reference clock in 1 MHz increments. Bits — Definition 00h — Timer disabled FFh-01h — <ClockRate> MHz

3.4 Device 1 Function 0 (Internal Graphics) Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space.

D1F0x00 Device/Vendor ID

Bits	Description
31:16	**DeviceID: device ID**. Read-only. Reset: Product-specific.
15:0	**VendorID: vendor ID**. Read-only. Reset: 1002h.

D1F0x04 Status/Command

Reset: 0010_0000h.

Bits	Description
31	**ParityErrorDetected: detected parity error**. Read; write-1-to-clear. 1=Poisoned TLP received.
30	**SignaledSystemError: signaled system error**. Read; write-1-to-clear. 1=A non-fatal or fatal error message was sent and SerrEn=1.
29	**ReceivedMasterAbort: received master abort**. Read; write-1-to-clear. 1=A completion with an unsupported request completion status was received.
28	**ReceivedTargetAbort: received target abort**. Read; write-1-to-clear. 1=A completion with completer abort completion status was received.
27	**SignalTargetAbort: Signaled target abort**. Read-only.
26:25	**DevselTiming: DEVSEL# Timing**. Read-only.
24	**MasterDataPerr: master data parity error**. Read; write-1-to-clear. 1=ParityErrorEn=1 and either a poisoned completion was received or the device poisoned a write request.
23	**FastBackCapable: fast back-to-back capable**. Read-only.
22	**UDFEn: UDF enable**. Read-only.
21	**PCI66En: 66 MHz capable**. Read-only.
20	**CapList: capability list**. Read-only. 1=capability list supported.
19	**IntStatus: interrupt status**. Read-only. 1=INTx interrupt message pending.
18:11	Reserved.
10	**IntDis: interrupt disable**. Read-write. 1=INTx interrupt messages generation disabled.
9	**FastB2BEn: fast back-to-back enable**. Read-only.
8	**SerrEn: System error enable**. Read-write. 1=Enables reporting of non-fatal and fatal errors detected.
7	**Stepping: Stepping control**. Read-only.
6	**ParityErrorEn: parity error response enable**. Read-write.
5	**PalSnoopEn: VGA palette snoop enable**. Read-only.
4	**MemWriteInvalidateEn: memory write and invalidate enable**. Read-only.
3	**SpecialCycleEn: special cycle enable**. Read-only.
2	**BusMasterEn: bus master enable**. Read-write. 1=Memory and IO read and write request generation enabled.

1	**MemAccessEn: IO access enable**. Read-write. This bit controls if memory accesses targeting this device are accepted. 1=Enabled. 0=Disabled.
0	**IoAccessEn: IO access enable**. Read-write. This bit controls if IO accesses targeting this device are accepted. 1=Enabled. 0=Disabled.

D1F0x08 Class Code/Revision ID

Bits	Description
31:8	**ClassCode**. Value: 03_0000h.
7:0	**RevID: revision ID**. Value: 00h.

D1F0x0C Header Type

Reset: 0000_0000h.

Bits	Description
31:24	**BIST**. Read-only.
23:16	**HeaderTypeReg**. Read-only. The header type field indicates a header type 0 and that this is a single function device.
15:8	**LatencyTimer**. Read-only. These bits are fixed at their default value.
7:0	**CacheLineSize**. Read-write. This field specifies the system cache line size in units of double words.

D1F0x10 Graphic Memory Base Address

IF (D0F0x64_x1C[F064BarEn]==0) THEN Reset: 0000_0008h. ELSE Reset: 0000_000Ch. ENDIF.

Bits	Description
31:26	**BaseAddr[31:26]: base address**. Read-write. The amount of memory requested by the graphics memory BAR is controlled by D0F0x64_x1C[MemApSize].
25:4	**BaseAddr[25:4]: base address**. Read-only.
3	**Pref: prefetchable**. Read-only. 1=Prefetchable memory region.
2:1	**Type: base address register type**. Read-only. 00b=32-bit BAR. 10b=64-bit BAR.
0	**MemSpace: memory space type**. Read-only. 0=Memory mapped base address.

IF (D0F0x64_x1C[F064BarEn]==0) THEN

D1F0x14 Graphics IO Base Address

Reset: 0000_0001h.

Bits	Description
31:8	**BaseAddr: base address**. IF (D0F0x64_x1C[IoBarDis]==0) THEN Read-write. ELSE Read-only. ENDIF.
7:1	Reserved.
0	**MemSpace: memory space type**. Read-only. 1=IO mapped base address.

ELSE

D1F0x14 Graphics Memory Base Address 64

Reset: 0000_0000h.

Bits	Description
31:0	**BaseAddr[63:32]: base address**. Read-write.

ENDIF.

D1F0x18 Graphics Memory Mapped Registers Base Address

IF (D0F0x64_x1C[F064BarEn]==0) THEN Reset: 0000_0000h. ELSE Reset: 0000_0004h. ENDIF.

Bits	Description
31:18	**BaseAddr[31:18]: base address**. Read-write.
17:16	**BaseAddr[17:16]: base address**. IF (D0F0x64_x1C[RegApSize]==0) THEN Read-write. ELSE Read-only. ENDIF.
15:4	**BaseAddr[15:4]: base address**. Read-only.
3	**Pref: prefetchable**. Read-only. 0=Non-prefetchable memory region.
2:1	**Type: base address register type**. Read-only. 00b=32-bit BAR. 10b=64-bit BAR.
0	**MemSpace: memory space type**. Read-only. 0=Memory mapped base address.

IF (D0F0x64_x1C[F064BarEn]==0) THEN

D1F0x1C Base Address 3

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

ELSE

D1F0x1C Graphics Memory Mapped Registers Address 64

Reset: 0000_0000h.

Bits	Description
31:0	**BaseAddr[63:32]: base address**. Read-write.

ENDIF.

IF (D0F0x64_x1C[F064BarEn]==0) THEN

D1F0x20 Base Address 4

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

ELSE

D1F0x20 Graphics IO Base Address

Reset: 0000_0000h.

Bits	Description
31:8	**BaseAddr: base address**. IF (D0F0x64_x1C[IoBarDis]==0) THEN Read-write. ELSE Read-only. ENDIF.
7:1	Reserved.
0	**MemSpace: memory space type**. Read-only. 1=IO mapped base address.

ENDIF.

D1F0x24 Base Address 5

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D1F0x2C Subsystem and Subvendor ID

Reset: 0000_0000h. This register can be modified through D1F0x4C

Bits	Description
31:16	**SubsystemID**. Read-only.
15:0	**SubsystemVendorID**. Read-only.

D1F0x30 Expansion ROM Base Address

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D1F0x34 Capabilities Pointer

Reset: 0000_0050h.

Bits	Description
31:8	Reserved.
7:0	**CapPtr: capabilities pointer**. Read-only. Pointer to PM capability.

D1F0x3C Interrupt Line

Reset: 0000_01FFh.

Bits	Description
31:11	Reserved.
10:8	**InterruptPin: interrupt pin**. Read-only. This field identifies the legacy interrupt message the function uses.
7:0	**InterruptLine: interrupt line**. Read-write. This field contains the interrupt line routing information.

D1F0x4C Subsystem and Subvendor ID Mirror

Reset: 0000_0000h.

Bits	Description
31:16	**SubsystemID**. Read-write. This field sets the value in the corresponding field in D1F0x2C.
15:0	**SubsystemVendorID**. Read-write. This field sets the value in the corresponding field in D1F0x2C.

D1F0x50 Power Management Capability

Bits	Description
31:27	**PmeSupport**. Value: 0. Indicates that PME is not supported.
26	**D2Support: D2 support**. Value: 1. Indicates that D2 is supported in hardware.
25	**D1Support: D1 support**. Value: 1. Indicates that D1 is supported in hardware.
24:22	**AuxCurrent: auxiliary current**. IF (D0F0xE4_x0101_0010[HwInitWrLock]==1) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0.
21	**DevSpecificInit: device specific initialization**. Value: 0. Indicates that there is no device specific initialization necessary.
20	Reserved.
19	**PmeClock**. Value: 0.
18:16	**Version: version**. Value: 011b.
15:8	**NextPtr: next pointer**. Value: 58h.The address of the next capability structure, or zero if this the end of the linked list of capability structures.
7:0	**CapID: capability ID**. Value: 01h. Indicates that the capability structure is a PCI power management data structure.

D1F0x54 Power Management Control and Status

Reset: 0000_0000h.

Bits	Description
31:24	**PmeData**. Read-only.
23	**BusPwrEn**. Read-only.

22	**B2B3Support**. Read-only. B states are not supported.
21:16	Reserved.
15	**PmeStatus: PME status**. Read-only.
14:13	**DataScale: data scale**. Read-only.
12:9	**DataSelect: data select**. Read-only.
8	**PmeEn: PME# enable**. Read-only.
7:4	Reserved.
3	**NoSoftReset: no soft reset**. Read-only. Software is required to re-initialize the function when returning from $D3_{hot}$.
2	Reserved.
1:0	**PowerState: power state**. Read-write. This 2-bit field is used both to determine the current power state of the root port and to set the root port into a new power state. Bits — Definition 00b — D0 10b-01b — Reserved 11b — D3hot

D1F0x58 PCI Express® Capability

Bits	Description
31:30	Reserved.
29:25	**IntMessageNum: interrupt message number**. Value: 0. This field indicates which MSI vector is used for the interrupt message.
24	**SlotImplemented: Slot implemented**. Value: 0.
23:20	**DeviceType: device type**. Value: IF (D0F0x64_x1C[RcieEn]==1) THEN 9. ELSEIF (D0F0x64_x1C[F0NonlegacyDeviceTypeEn]==0) THEN 1. ELSE 0. ENDIF. 0=PCIe® endpoint. 1=Legacy PCIe® endpoint. 9=Root complex integrated endpoint.
19:16	**Version**. Value: 2h.
15:8	**NextPtr: next pointer**. Value: IF (D0F0x64_x1C[MsiDis]==0) THEN A0h. ELSE 00h. ENDIF. The address of the next capability structure, or zero if this the end of the linked list of capability structures.
7:0	**CapID: capability ID**. Value: 10h. Indicates a PCIe® Capability structure.

D1F0x5C Device Capability

Bits	Description
31:29	Reserved.
28	**FlrCapable: function level reset capability**. Value: 0.
27:26	**CapturedSlotPowerScale: captured slot power limit scale**. Value: 0.
25:18	**CapturedSlotPowerLimit: captured slot power limit value**. Value: 0.
17:16	Reserved.

15	**RoleBasedErrReporting: role-based error reporting**. Value: 1.
14:12	Reserved.
11:9	**L1AcceptableLatency: endpoint L1 Acceptable Latency**. Value: 111b.
8:6	**L0SAcceptableLatency: endpoint L0s Acceptable Latency**. Value: 110b.
5	**ExtendedTag: extended tag support**. Value: 1. Indicates 8 bit tag support.
4:3	**PhantomFunc: phantom function support**. Value: 0. Indicates no phantom functions supported.
2:0	**MaxPayloadSupport: maximum supported payload size**. Value: 0. 128 bytes max payload size.

D1F0x60 Device Control and Status

Reset: 0000_0810h.

Bits	Description
31:22	Reserved.
21	**TransactionsPending: transactions pending**. Read-only.
20	**AuxPwr: auxiliary power**. Read-only.
19	**UsrDetected: unsupported request detected**. Read; set-by-hardware; write-1-to-clear. 1=Unsupported request received.
18	**FatalErr: fatal error detected**. Read; set-by-hardware; write-1-to-clear. 1=Fatal error detected.
17	**NonFatalErr: non-fatal error detected**. Read; set-by-hardware; write-1-to-clear. 1=Non-fatal error detected.
16	**CorrErr: correctable error detected**. Read; set-by-hardware; write-1-to-clear. 1=Correctable error detected.
15	**BridgeCfgRetryEn: bridge configuration retry enable**. Read-only.
14:12	**MaxRequestSize: maximum request size**. Read-only.
11	**NoSnoopEnable: enable no snoop**. Read-write. 1=The device is permitted to set the No Snoop bit in requests.
10	**AuxPowerPmEn: auxiliary power PM enable**. Read-only. This capability is not implemented.
9	**PhantomFuncEn: phantom functions enable**. Read-only. Phantom functions are not supported.
8	**ExtendedTagEn: extended tag enable**. Read-write. 1=8-bit tag request tags. 0=5-bit request tag.
7:5	**MaxPayloadSize: maximum supported payload size**. Read-only. 000b=Indicates a 128 byte maximum payload size.
4	**RelaxedOrdEn: relaxed ordering enable**. Read-write. 1=The device is permitted to set the Relaxed Ordering bit.
3	**UsrReportEn: unsupported request reporting enable**. Read-write. 1=Enables signaling unsupported requests by sending error messages.
2	**FatalErrEn: fatal error reporting enable**. Read-write. 1=Enables sending ERR_FATAL message when a fatal error is detected.
1	**NonFatalErrEn: non-fatal error reporting enable**. Read-write. 1=Enables sending ERR_NONFATAL message when a non-fatal error is detected.
0	**CorrErrEn: correctable error reporting enable**. Read-write. 1=Enables sending ERR_CORR message when a correctable error is detected.

D1F0x64 Link Capability

Bits	Description
31:24	**PortNumber: port number**. Value: 0. This field indicates the PCI Express port number for the given PCI Express link.
23:22	Reserved.
21	**LinkBWNotificationCap: link bandwidth notification capability**. Value: 0.
20	**DlActiveReportingCapable: data link layer active reporting capability**. Value: 0.
19	**SurpriseDownErrReporting: surprise down error reporting capability**. Value: 0.
18	**ClockPowerManagement: clock power management**. Value: 0. Indicates that the reference clock must not be removed while in L1 or L2/L3 ready link states.
17:15	**L1ExitLatency: L1 exit latency**. Value: 0.
14:12	**L0sExitLatency: L0s exit latency**. Value: 0.
11:10	**PMSupport: active state power management support**. Value: 11b.
9:4	**LinkWidth: maximum link width**. Value: 0.
3:0	**LinkSpeed: link speed**. Value: 0.

D1F0x68 Link Control and Status

Reset: 1000_0000h.

Bits	Description
31	**LinkAutonomousBWStatus: link autonomous bandwidth status**. Read-only.
30	**LinkBWManagementStatus: link bandwidth management status**. Read-only.
29	**DlActive: data link layer link active**. Read-only. This bit indicates the status of the data link control and management state machine. Reads return a 1 to indicate the DL_Active state, otherwise 0 is returned.
28	**SlotClockCfg: slot clock configuration**. Read-only. 1=The root port uses the same clock that the platform provides.
27	**LinkTraining: link training**. Read-only. 1=Indicates that the physical layer link training state machine is in the configuration or recovery state, or that 1b was written to the RetrainLink bit but link training has not yet begun. Hardware clears this bit when the link training state machine exits the configuration/recovery state.
26	Reserved.
25:20	**NegotiatedLinkWidth: negotiated link width**. Read-only. This field indicates the negotiated width of the given PCI Express link.
19:16	**LinkSpeed: link speed**. Read-only.
15:12	Reserved.
11	**LinkAutonomousBWIntEn: link autonomous bandwidth interrupt enable**. Read-only.
10	**LinkBWManagementEn: link bandwidth management interrupt enable**. Read-only.

9	**HWAutonomousWidthDisable: hardware autonomous width disable**. Read-write. 1=Hardware not allowed to change the link width except to correct unreliable link operation by reducing link width.
8	**ClockPowerManagementEn: clock power management enable**. Read-write.
7	**ExtendedSync: extended sync**. Read-write. 1=Forces the transmission of additional ordered sets when exiting the L0s state and when in the recovery state.
6	**CommonClockCfg: common clock configuration**. Read-write. 1=Indicates that the root port and the component at the opposite end of this Link are operating with a distributed common reference clock. 0=Indicates that the upstream port and the component at the opposite end of this Link are operating with asynchronous reference clock.
5	**RetrainLink: retrain link**. Read-only. This bit does not apply to endpoints.
4	**LinkDis: link disable**. Read-only. This bit does not apply to endpoints.
3	**ReadCplBoundary: read completion boundary**. Read-only. 0=64 byte read completion boundary.
2	Reserved.
1:0	**PmControl: active state power management enable**. Read-write. This field controls the level of ASPM supported on the given PCI Express link. Bits Definition Bits Definition 00b Disabled. 10b L1 Entry Enabled. 01b L0s Entry Enabled. 11b L0s and L1 Entry Enabled.

D1F0x7C Device Capability 2

Reset: 0000_0000h.

Bits	Description
31:5	Reserved.
4	**CplTimeoutDisSup: completion timeout disable supported**. Read-only.
3:0	**CplTimeoutRangeSup: completion timeout range supported**. Read-only.

D1F0x80 Device Control and Status 2

Reset: 0000_0000h.

Bits	Description
31:5	Reserved.
4	**CplTimeoutDis: completion timeout disable**. Read-only.
3:0	**CplTimeoutValue: completion timeout range supported**. Read-only.

D1F0x84 Link Capability 2

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D1F0x88 Link Control and Status 2

Reset: 0000_0000h.

Bits	Description
31:17	Reserved.
16	**CurDeemphasisLevel: current de-emphasis level**. Read-only. 1=-3.5 dB. 0=-6 dB.
15:13	Reserved.
12	**ComplianceDeemphasis: compliance de-emphasis**. Read-write. This bit defines the de-emphasis level used in compliance mode. 1=-3.5 dB. 0=-6 dB.
11	**ComplianceSOS: compliance SOS**. Read-write. 1=The device transmits skip ordered sets in between the modified compliance pattern.
10	**EnterModCompliance: enter modified compliance**. Read-write. 1=The device transmits modified compliance pattern.
9:7	**XmitMargin: transmit margin**. Read-write. This field controls the voltage level (without de-emphasis) at the transmitter pins.
6	**SelectableDeemphasis: selectable de-emphasis**. Read-only. 1=Selectable de-emphasis is supported. 0=Selectable de-emphasis is not supported.
5	**HwAutonomousSpeedDisable: hardware autonomous speed disable**. Read-write. 1=Disables hardware generated link speed changes.
4	**EnterCompliance: enter compliance**. Read-write. 1=Force link to enter compliance mode.
3:0	**TargetLinkSpeed: target link speed**. Read-write. This field defines the upper limit of the link operational speed.

D1F0xA0 MSI Capability

Bits	Description
31:24	Reserved.
23	**Msi64bit: MSI 64 bit capability**. Read-only. Value: D0F0x64_x46[Msi64bitEn]. 1=The device is capable of sending 64-bit MSI messages. 0=The device is not capable of sending a 64-bit message address.
22:20	**MsiMultiEn: MSI multiple message enable**. Read-write. Reset: 000b. Software writes to this field to indicate the number of allocated vectors (equal to or less than the number of requested vectors). When MSI is enabled, a function is allocated at least 1 vector.
19:17	**MsiMultiCap: MSI multiple message capability**. Read-only. Reset: 000b. 000b=The device is requesting one vector.
16	**MsiEn: MSI enable**. Read-write. Reset: 0. 1=MSI generation is enabled and INTx generation is disabled. 0=MSI generation disabled and INTx generation is enabled.
15:8	**NextPtr: next pointer**. Read-only. Reset: 00h. The address of the next capability structure, or zero if this the end of the linked list of capability structures.
7:0	**CapID: capability ID**. Read-only. Reset: 05h. 05h=MSI capability structure.

D1F0xA4 MSI Message Address Low

Reset: 0000_0000h.

Bits	Description
31:2	**MsiMsgAddrLo: MSI message address**. Read-write. This register specifies the doubleword aligned address for the MSI memory write transaction.
1:0	Reserved.

IF (D0F0x64_x46[Msi64bitEn]==1) THEN

D1F0xA8 MSI Message Address High

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**MsiMsgAddrHi: MSI message address**. Read-write. This register specifies the upper 8 bits of the MSI address in 64-bit MSI mode.

ENDIF.

IF (D0F0x64_x46[Msi64bitEn]==0) THEN

D1F0xA8 MSI Message Data

ELSEIF (D0F0x64_x46[Msi64bitEn]==1) THEN

D1F0xAC MSI Message Data

ENDIF.
Reset: 0000_0000h.

Bits	Description
31:16	Reserved.
15:0	**MsiData: MSI message data**. Read-write. This register specifies lower 16 bits of data for the MSI memory write transaction. The upper 16 bits are always 0.

D1F0x100 Vendor Specific Enhanced Capability

Reset: 0001_000Bh.

Bits	Description
31:20	**NextPtr: next pointer**. Read-only. The address of the next capability structure, or zero if this the end of the linked list of capability structures.
19:16	**CapVer: capability version**. Read-only.
15:0	**CapID: capability ID**. Read-only.

D1F0x104 Vendor Specific Header

Reset: 0101_0001h.

Bits	Description
31:20	**VsecLen: vendor specific enhanced next pointer**. Read-only.
19:16	**VsecRev: vendor specific enhanced capability version**. Read-only.
15:0	**VsecID: vendor specific enhanced capability ID**. Read-only.

D1F0x108 Vendor Specific 1

Reset: 0000_0000h.

Bits	Description
31:0	**Scratch: scratch**. Read-write.

D1F0x10C Vendor Specific 2

Reset: 0000_0000h.

Bits	Description
31:0	**Scratch: scratch**. Read-write.

3.5 Device 1 Function 1 (Audio Controller) Configuration Registers

Access to the internal graphic configuration registers requires that internal graphic bridge is configured to forward configuration transactions to the graphics bus. See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space.

D1F1x00 Device/Vendor ID

Bits	Description
31:16	**DeviceID: device ID**. Read-only. Reset: 1314h.
15:0	**VendorID: vendor ID**. Read-only. Reset: 1002h.

D1F1x04 Status/Command

Reset: 0010_0000h.

Bits	Description
31	**ParityErrorDetected: detected parity error**. Read; write-1-to-clear. 1=Poisoned TLP received.
30	**SignaledSystemError: signaled system error**. Read; write-1-to-clear. 1=A non-fatal or fatal error message was sent and SerrEn=1.
29	**ReceivedMasterAbort: received master abort**. Read; write-1-to-clear. 1=A completion with an unsupported request completion status was received.

28	**ReceivedTargetAbort: received target abort**. Read; write-1-to-clear. 1=A completion with completer abort completion status was received.
27	**SignalTargetAbort: Signaled target abort**. Read-only.
26:25	**DevselTiming: DEVSEL# Timing**. Read-only.
24	**MasterDataPerr: master data parity error**. Read; write-1-to-clear. 1=ParityErrorEn=1 and either a poisoned completion was received or the device poisoned a write request.
23	**FastBackCapable: fast back-to-back capable**. Read-only.
22	**UDFEn: UDF enable**. Read-only.
21	**PCI66En: 66 MHz capable**. Read-only.
20	**CapList: capability list**. Read-only. 1=capability list supported.
19	**IntStatus: interrupt status**. Read-only. 1=INTx interrupt message pending.
18:11	Reserved.
10	**IntDis: interrupt disable**. Read-write. 1=INTx interrupt messages generation disabled.
9	**FastB2BEn: fast back-to-back enable**. Read-only.
8	**SerrEn: System error enable**. Read-write. 1=Enables reporting of non-fatal and fatal errors detected.
7	**Stepping: Stepping control**. Read-only.
6	**ParityErrorEn: parity error response enable**. Read-write.
5	**PalSnoopEn: VGA palette snoop enable**. Read-only.
4	**MemWriteInvalidateEn: memory write and invalidate enable**. Read-only.
3	**SpecialCycleEn: special cycle enable**. Read-only.
2	**BusMasterEn: bus master enable**. Read-write. 1=Memory and IO read and write request generation enabled.
1	**MemAccessEn: IO access enable**. Read-write. This bit controls if memory accesses targeting this device are accepted. 1=Enabled. 0=Disabled.
0	**IoAccessEn: IO access enable**. Read-write. This bit controls if IO accesses targeting this device are accepted. 1=Enabled. 0=Disabled.

D1F1x08 Class Code/Revision ID

Reset: 0403_0000h.

Bits	Description
31:8	**ClassCode**. Read-only.
7:0	**RevID: revision ID**. Read-only.

D1F1x0C Header Type

Reset: 0080_0000h.

Bits	Description
31:24	**BIST**. Read-only. These bits are fixed at their default values.
23:16	**HeaderTypeReg**. Read-only. 80h=Type 0 multi-function device.

15:8	**LatencyTimer**. Read-only. These bits are fixed at their default value.
7:0	**CacheLineSize**. Read-write. This field specifies the system cache line size in units of double words.

D1F1x10 Audio Registers Base Address

Reset: 0000_0000h.

Bits	Description
31:14	**BaseAddr: base address**. Read-write.
13:4	Reserved.
3	**Pref: prefetchable**. Read-only. 0=Non-prefetchable memory region.
2:1	**Type: base address register type**. Read-only. 00b=32-bit base address register.
0	**MemSpace: memory space type**. Read-only. 0=Memory mapped base address.

D1F1x14 Base Address 1

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D1F1x18 Base Address 2

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D1F1x1C Base Address 3

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D1F1x20 Base Address 4

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D1F1x24 Base Address 5

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D1F1x2C Subsystem and Subvendor ID

Reset: 0000_0000h. This register can be modified through D1F1x4C.

Bits	Description
31:16	**SubsystemID**. Read-only.
15:0	**SubsystemVendorID**. Read-only.

D1F1x30 Expansion ROM Base Address

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D1F1x34 Capabilities Pointer

Reset: 0000_0050h.

Bits	Description
31:8	Reserved.
7:0	**CapPtr: capabilities pointer**. Read-only. Pointer to PM capability.

D1F1x3C Interrupt Line

Reset: 0000_02FFh.

Bits	Description
31:11	Reserved.
10:8	**InterruptPin: interrupt pin**. Read-only. This field identifies the legacy interrupt message the function uses.
7:0	**InterruptLine: interrupt line**. Read-write. This field contains the interrupt line routing information.

D1F1x4C Subsystem and Subvendor ID Mirror

Reset: 0000_0000h.

Bits	Description
31:16	**SubsystemID**. Read-write. This field sets the value in the corresponding field in D1F1x2C.
15:0	**SubsystemVendorID**. Read-write. This field sets the value in the corresponding field in D1F1x2C.

D1F1x50 Power Management Capability

Bits	Description
31:27	**PmeSupport**. Value: 0. Indicates that there is no PME support.
26	**D2Support: D2 support**. Value: 1. Indicates that D2 is supported in hardware.
25	**D1Support: D1 support**. Value: 1. Indicates that D1 is supported in hardware.
24:22	**AuxCurrent: auxiliary current**. IF (D0F0xE4_x0101_0010[HwInitWrLock]==1) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0.
21	**DevSpecificInit: device specific initialization**. Value: 0. Indicates that there is no device specific initialization necessary.
20	Reserved.
19	**PmeClock**. Value: 0.
18:16	**Version: version**. Value: 011b.
15:8	**NextPtr: next pointer**. Value: 58h. The address of the next capability structure, or zero if this the end of the linked list of capability structures.
7:0	**CapID: capability ID**. Value: 01h. Indicates that the capability structure is a PCI power management data structure.

D1F1x54 Power Management Control and Status

Reset: 0000_0000h.

Bits	Description
31:24	**PmeData**. Read-only.
23	**BusPwrEn**. Read-only.
22	**B2B3Support**. Read-only. B states are not supported.
21:16	Reserved.
15	**PmeStatus: PME status**. Read-only.
14:13	**DataScale: data scale**. Read-only.
12:9	**DataSelect: data select**. Read-only.
8	**PmeEn: PME# enable**. Read-only.
7:4	Reserved.

3	**NoSoftReset: no soft reset**. Read-only. Software is required to re-initialize the function when returning from $D3_{hot}$.
2	Reserved.
1:0	**PowerState: power state**. Read-write. This 2-bit field is used both to determine the current power state of the root port and to set the root port into a new power state. Bits Definition 00b D0 11b D3hot

D1F1x58 PCI Express Capability

Bits	Description
31:30	Reserved.
29:25	**IntMessageNum: interrupt message number**. Value: 0. This field indicates which MSI vector is used for the interrupt message.
24	**SlotImplemented: Slot implemented**. Value: 0.
23:20	**DeviceType: device type**. Value: IF (D0F0x64_x1C[AudioNonlegacyDeviceTypeEn]==0) THEN 1. ELSE 0. ENDIF. 0=PCIe® endpoint.
19:16	**Version**. Value: 2h.
15:8	**NextPtr: next pointer**. Value: IF (D0F0x64_x1C[MsiDis]==0) THEN A0h. ELSE 00h. ENDIF. The address of the next capability structure, or zero if this the end of the linked list of capability structures.
7:0	**CapID: capability ID**. Value: 10h. Indicates a PCIe® Capability structure.

D1F1x5C Device Capability

Bits	Description
31:29	Reserved.
28	**FlrCapable: function level reset capability**. Value: 0.
27:26	**CapturedSlotPowerScale: captured slot power limit scale**. Value: 0.
25:18	**CapturedSlotPowerLimit: captured slot power limit value**. Value: 0.
17:16	Reserved.
15	**RoleBasedErrReporting: role-based error reporting**. Value: 1.
14:12	Reserved.
11:9	**L1AcceptableLatency: endpoint L1 Acceptable Latency**. Value: 111b.
8:6	**L0SAcceptableLatency: endpoint L0s Acceptable Latency**. Value: 110b.
5	**ExtendedTag: extended tag support**. Value: 1. Indicates 8 bit tag support.
4:3	**PhantomFunc: phantom function support**. Value: 0. Indicates no phantom functions supported.
2:0	**MaxPayloadSupport: maximum supported payload size**. Value: 0. Indicates 128 bytes max payload size.

D1F1x60 Device Control and Status

Reset: 0000_0810h.

Bits	Description
31:22	Reserved.
21	**TransactionsPending: transactions pending**. Read-only.
20	**AuxPwr: auxiliary power**. Read-only.
19	**UsrDetected: unsupported request detected**. Read; set-by-hardware; write-1-to-clear. 1=Unsupported request received.
18	**FatalErr: fatal error detected**. Read; set-by-hardware; write-1-to-clear. 1=Fatal error detected.
17	**NonFatalErr: non-fatal error detected**. Read; set-by-hardware; write-1-to-clear. 1=Non-fatal error detected.
16	**CorrErr: correctable error detected**. Read; set-by-hardware; write-1-to-clear. 1=Correctable error detected.
15	**BridgeCfgRetryEn: bridge configuration retry enable**. Read-only.
14:12	**MaxRequestSize: maximum request size**. Read-only. 0=The root port never generates read requests with size exceeding 128 bytes.
11	**NoSnoopEnable: enable no snoop**. Read-write. 1=The device is permitted to set the No Snoop bit in requests.
10	**AuxPowerPmEn: auxiliary power PM enable**. Read-only. This capability is not implemented.
9	**PhantomFuncEn: phantom functions enable**. Read-only. Phantom functions are not supported.
8	**ExtendedTagEn: extended tag enable**. Read-write. 1=8-bit tag request tags. 0=5-bit request tag.
7:5	**MaxPayloadSize: maximum supported payload size**. Read-only. 000b=Indicates a 128 byte maximum payload size.
4	**RelaxedOrdEn: relaxed ordering enable**. Read-write. 1=The device is permitted to set the Relaxed Ordering bit.
3	**UsrReportEn: unsupported request reporting enable**. Read-write. 1=Enables signaling unsupported requests by sending error messages.
2	**FatalErrEn: fatal error reporting enable**. Read-write. 1=Enables sending ERR_FATAL message when a fatal error is detected.
1	**NonFatalErrEn: non-fatal error reporting enable**. Read-write. 1=Enables sending ERR_NONFATAL message when a non-fatal error is detected.
0	**CorrErrEn: correctable error reporting enable**. Read-write. 1=Enables sending ERR_CORR message when a correctable error is detected.

D1F1x64 Link Capability

Bits	Description
31:24	**PortNumber: port number**. Value: 0. This field indicates the PCI Express port number for the given PCI Express link.
23:22	Reserved.

21	**LinkBWNotificationCap: link bandwidth notification capability**. Value: 0.
20	**DlActiveReportingCapable: data link layer active reporting capability**. Value: 0.
19	**SurpriseDownErrReporting: surprise down error reporting capability**. Value: 0.
18	**ClockPowerManagement: clock power management**. Value: 0. Indicates that the reference clock must not be removed while in L1 or L2/L3 ready link states.
17:15	**L1ExitLatency: L1 exit latency**. Value: 0.
14:12	**L0sExitLatency: L0s exit latency**. Value: 0.
11:10	**PMSupport: active state power management support**. Value: 11b.
9:4	**LinkWidth: maximum link width**. Value: 0.
3:0	**LinkSpeed: link speed**. Value: 0.

D1F1x68 Link Control and Status

Reset: 1000_0000h.

Bits	Description
31	**LinkAutonomousBWStatus: link autonomous bandwidth status**. Read-only.
30	**LinkBWManagementStatus: link bandwidth management status**. Read-only.
29	**DlActive: data link layer link active**. Read-only. This bit indicates the status of the data link control and management state machine. Reads return a 1 to indicate the DL_Active state, otherwise 0 is returned.
28	**SlotClockCfg: slot clock configuration**. Read-only. 1=The root port uses the same clock that the platform provides.
27	**LinkTraining: link training**. Read-only. 1=Indicates that the physical layer link training state machine is in the configuration or recovery state, or that 1b was written to the RetrainLink bit but link training has not yet begun. Hardware clears this bit when the link training state machine exits the configuration/recovery state.
26	Reserved.
25:20	**NegotiatedLinkWidth: negotiated link width**. Read-only. This field indicates the negotiated width of the given PCI Express link.
19:16	**LinkSpeed: link speed**. Read-only. 0001b: 2.5 Gb/s. 0010b: 5 Gb/s
15:12	Reserved.
11	**LinkAutonomousBWIntEn: link autonomous bandwidth interrupt enable**. Read-only.
10	**LinkBWManagementEn: link bandwidth management interrupt enable**. Read-only.
9	**HWAutonomousWidthDisable: hardware autonomous width disable**. Read-write. 1=Hardware not allowed to change the link width except to correct unreliable link operation by reducing link width.
8	**ClockPowerManagementEn: clock power management enable**. Read-write.
7	**ExtendedSync: extended sync**. Read-write. 1=Forces the transmission of additional ordered sets when exiting the L0s state and when in the recovery state.

6	**CommonClockCfg: common clock configuration**. Read-write. 1=Indicates that the root port and the component at the opposite end of this Link are operating with a distributed common reference clock. 0=Indicates that the upstream port and the component at the opposite end of this Link are operating with asynchronous reference clock.
5	**RetrainLink: retrain link**. Read-only. This bit does not apply to endpoints.
4	**LinkDis: link disable**. Read-only. This bit does not apply to endpoints.
3	**ReadCplBoundary: read completion boundary**. Read-only. 0=64 byte read completion boundary.
2	Reserved.
1:0	**PmControl: active state power management enable**. Read-write. This field controls the level of ASPM supported on the given PCI Express link. Bits Definition Bits Definition 00b Disabled. 10b L1 Entry Enabled. 01b L0s Entry Enabled. 11b L0s and L1 Entry Enabled.

D1F1x7C Device Capability 2

Reset: 0000_0000h.

Bits	Description
31:5	Reserved.
4	**CplTimeoutDisSup: completion timeout disable supported**. Read-only.
3:0	**CplTimeoutRangeSup: completion timeout range supported**. Read-only.

D1F1x80 Device Control and Status 2

Reset: 0000_0000h.

Bits	Description
31:5	Reserved.
4	**CplTimeoutDis: completion timeout disable**. Read-only.
3:0	**CplTimeoutValue: completion timeout range supported**. Read-only.

D1F1x84 Link Capability 2

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D1F1x88 Link Control and Status 2

Reset: 0000_0000h.

Bits	Description
31:17	Reserved.

16	**CurDeemphasisLevel: current de-emphasis level**. Read-only. 1=-3.5 dB. 0=-6 dB.
15:13	Reserved.
12	**ComplianceDeemphasis: compliance de-emphasis**. Read-write. This bit defines the de-emphasis level used in compliance mode. 1=-3.5 dB. 0=-6 dB.
11	**ComplianceSOS: compliance SOS**. Read-write. 1=The device transmits skip ordered sets in between the modified compliance pattern.
10	**EnterModCompliance: enter modified compliance**. Read-write. 1=The device transmits modified compliance pattern.
9:7	**XmitMargin: transmit margin**. Read-write. This field controls the voltage level (without de-emphasis) at the transmitter pins.
6	**SelectableDeemphasis: selectable de-emphasis**. Read-only. 1=Selectable de-emphasis is supported. 0=Selectable de-emphasis is not supported.
5	**HwAutonomousSpeedDisable: hardware autonomous speed disable**. Read-write. 1=Disables hardware generated link speed changes.
4	**EnterCompliance: enter compliance**. Read-write. 1=Force link to enter compliance mode.
3:0	**TargetLinkSpeed: target link speed**. Read-write. This field defines the upper limit of the link operational speed.

D1F1xA0 PCI Express MSI Capability

Bits	Description
31:24	Reserved.
23	**Msi64bit: MSI 64 bit capability**. Read-only. Value: D0F0x64_x46[Msi64bitEn]. 1=The device is capable of sending 64-bit MSI messages. 0=The device is not capable of sending a 64-bit message address.
22:20	**MsiMultiEn: MSI multiple message enable**. Read-write. Reset: 000b. Software writes to this field to indicate the number of allocated vectors (equal to or less than the number of requested vectors). When MSI is enabled, a function is allocated at least 1 vector.
19:17	**MsiMultiCap: MSI multiple message capability**. Read-only. Reset: 000b. 000b=The device is requesting one vector.
16	**MsiEn: MSI enable**. Read-write. Reset: 0. 1=MSI generation is enabled and INTx generation is disabled. 0=MSI generation disabled and INTx generation is enabled.
15:8	**NextPtr: next pointer**. Read-only. Reset: 00h. The address of the next capability structure, or zero if this the end of the linked list of capability structures.
7:0	**CapID: capability ID**. Read-only. Reset: 05h. 05h=MSI capability structure.

D1F1xA4 MSI Message Address Low

Reset: 0000_0000h.

Bits	Description
31:2	**MsiMsgAddrLo: MSI message address**. Read-write. This register specifies the doubleword aligned address for the MSI memory write transaction.
1:0	Reserved.

IF (D0F0x64_x46[Msi64bitEn]==1) THEN

D1F1xA8 MSI Message Address High

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**MsiMsgAddrHi: MSI message address**. Read-write. This register specifies the upper 8 bits of the MSI address in 64-bit MSI mode.

ENDIF.

IF (D0F0x64_x46[Msi64bitEn]==0) THEN

D1F1xA8 MSI Message Data

ELSEIF (D0F0x64_x46[Msi64bitEn]==1) THEN

D1F1xAC MSI Message Data

ENDIF.
Reset: 0000_0000h.

Bits	Description
31:16	Reserved.
15:0	**MsiData: MSI message data**. Read-write. This register specifies lower 16 bits of data for the MSI memory write transaction. The upper 16 bits are always 0.

D1F1x100 Vendor Specific Enhanced Capability

Reset: 0001_000Bh.

Bits	Description
31:20	**NextPtr: next pointer**. Read-only. The address of the next capability structure, or zero if this the end of the linked list of capability structures.
19:16	**CapVer: capability version**. Read-only.
15:0	**CapID: capability ID**. Read-only.

D1F1x104 Vendor Specific Header

Reset: 0101_0001h.

Bits	Description
31:20	**VsecLen: vendor specific enhanced next pointer**. Read-only.
19:16	**VsecRev: vendor specific enhanced capability version**. Read-only.
15:0	**VsecID: vendor specific enhanced capability ID**. Read-only.

D1F1x108 Vendor Specific 1

Reset: 0000_0000h.

Bits	Description
31:0	**Scratch: scratch**. Read-write.

D1F1x10C Vendor Specific 2

Reset: 0000_0000h.

Bits	Description
31:0	**Scratch: scratch**. Read-write.

3.6 Device [8:4] Function 0 (Root Port) Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space.

D[8:4]F0x00 Device/Vendor ID

Table 50: Reset mapping for D[8:4]F0x00

Register	Reset
D4F0x00	1512_1022h.
D5F0x00	1513_1022h.
D6F0x00	1514_1022h.
D7F0x00	1515_1022h.
D8F0x00	1516_1022h.

Bits	Description
31:16	**DeviceID: device ID**. Read-only.
15:0	**VendorID: vendor ID**. Read-only.

D[8:4]F0x04 Status/Command

Reset: 0010_0000h.

Bits	Description
31	**ParityErrorDetected: detected parity error**. Read; set-by-hardware; write-1-to-clear.
30	**SignaledSystemError: signaled system error**. Read; set-by-hardware; write-1-to-clear. 1=System error signaled.
29	**ReceivedMasterAbort: received master abort**. Read; set-by-hardware; write-1-to-clear.
28	**ReceivedTargetAbort: received target abort**. Read; set-by-hardware; write-1-to-clear.
27	**SignalTargetAbort: signaled target abort**. Read; set-by-hardware; write-1-to-clear.
26:25	**DevselTiming: DEVSEL# Timing**. Read-only.
24	**DataPerr: data parity error**. Read; set-by-hardware; write-1-to-clear.
23	**FastBackCapable: fast back-to-back capable**. Read-only.
22	**UDFEn: UDF enable**. Read-only.
21	**PCI66En: 66 MHz capable**. Read-only.
20	**CapList: capability list**. Read-only. 1= Capability list present.
19	**IntStatus: interrupt status**. Read-only. 1=An INTx interrupt Message is pending in the device.
18:11	Reserved.
10	**IntDis: interrupt disable**. Read-write.
9	**FastB2BEn: fast back-to-back enable**. Read-only.
8	**SerrEn: system error enable**. Read-write. 1=System error reporting enabled.
7	**Stepping: Stepping control**. Read-only.
6	**ParityErrorEn: parity error response enable**. Read-write.
5	**PalSnoopEn: VGA palette snoop enable**. Read-only.
4	**MemWriteInvalidateEn: memory write and invalidate enable**. Read-only.
3	**SpecialCycleEn: special cycle enable**. Read-only.
2	**BusMasterEn: bus master enable**. Read-write.
1	**MemAccessEn: IO access enable**. Read-write. This bit controls if memory accesses targeting this device are accepted or not. 1=Enabled. 0=Disabled.
0	**IoAccessEn: IO access enable**. Read-write. This bit controls if IO accesses targeting this device are accepted or not. 1=Enabled. 0=Disabled.

D[8:4]F0x08 Class Code/Revision ID

Reset: 0604_00xxh.

Bits	Description
31:8	**ClassCode**. Read-only. Provides the host bridge class code as defined in the PCI specification.
7:0	**RevID: revision ID**. Read-only.

D[8:4]F0x0C Header Type

Reset: 0001_0000h.

Bits	Description
31:24	**BIST**. Read-only. These bits are fixed at their default values.
23:16	**HeaderTypeReg**. Read-only. These bits are fixed at their default values. The header type field indicates a header type 1 and that there is only one function present in this device.
15:8	**LatencyTimer**. Read-only. This field does not control any hardware.
7:0	**CacheLineSize**. Read-write.

D[8:4]F0x18 Bus Number and Secondary Latency

Reset: 0000_0000h.

Bits	Description
31:24	**SecondaryLatencyTimer: secondary latency timer**. Read-only. This field is always 0.
23:16	**SubBusNumber: subordinate number**. Read-write. This field contains the highest-numbered bus that exists on the secondary side of the bridge.
15:8	**SecondaryBus: secondary bus number**. Read-write. This field defines the bus number of the secondary bus interface.
7:0	**PrimaryBus: primary bus number**. Read-write. This field defines the bus number of the primary bus interface.

D[8:4]F0x1C IO Base and Secondary Status

Reset: 0000_0101h.

Bits	Description
31	**ParityErrorDetected: detected parity error**. Read; set-by-hardware; write-1-to-clear. A Poisoned TLP was received regardless of the state of the D[8:4]F0x04[ParityErrorEn].
30	**ReceivedSystemError: signaled system error**. Read; set-by-hardware; write-1-to-clear. 1=A System Error was detected.
29	**ReceivedMasterAbort: received master abort**. Read; set-by-hardware; write-1-to-clear. 1=A CPU transaction is terminated due to a master-abort.
28	**ReceivedTargetAbort: received target abort**. Read; set-by-hardware; write-1-to-clear. 1=A CPU transaction (except for a special cycle) is terminated due to a target-abort.
27	**SignalTargetAbort: signaled target abort**. Read; set-by-hardware; write-1-to-clear.
26:25	**DevselTiming: DEVSEL# Timing**. Read-only.
24	**MasterDataPerr: master data parity error**. Read; set-by-hardware; write-1-to-clear. 1=The link received a poisoned or poisoned a downstream write and D[8:4]F0x3C[ParityResponseEn]=1.
23	**FastBackCapable: fast back-to-back capable**. Read-only.
22	**UDFEn: UDF enable**. Read-only.
21	**PCI66En: 66 MHz capable**. Read-only.
20	**CapList: capability list**. Read-only.

19:16	Reserved.
15:12	**IOLimit[15:12]**. Read-write. Lower part of the limit address. Upper part is defined in D[8:4]F0x30.
11:8	Reserved.
7:4	**IOBase[15:12]**. Read-write. Lower part of the base address. Upper part is defined in D[8:4]F0x30.
3:0	Reserved.

D[8:4]F0x20 Memory Limit and Base

Reset: 0000_0000h.

Bits	Description
31:20	**MemLimit**. Read-write.
19:16	Reserved.
15:4	**MemBase**. Read-write.
3:0	Reserved.

D[8:4]F0x24 Prefetchable Memory Limit and Base

Reset: 0001_0001h.

Bits	Description
31:20	**PrefMemLimit**. Read-write. Lower part of the limit address. Upper part is defined in D[8:4]F0x2C.
19:16	**PrefMemLimitR**. Read-only. 1=64 bit memory address decoder.
15:4	**PrefMemBase[31:20]**. Read-write. Lower part of the base address. Upper part is defined in D[8:4]F0x28.
3:0	**PrefMemBaseR**. Read-only. 1= 64 bit memory address decoder.

D[8:4]F0x28 Prefetchable Memory Base High

Reset: 0000_0000h.

Bits	Description
31:0	**PrefMemBase[63:32]**. Read-write. Upper part of the base address. Lower part is defined in D[8:4]F0x24.

D[8:4]F0x2C Prefetchable Memory Limit High

Reset: 0000_0000h.

Bits	Description
31:0	**PrefMemLimit[63:32]**. Read-write. Upper part of the limit address. Lower part is defined in D[8:4]F0x24.

D[8:4]F0x30 IO Base and Limit High

Reset: 0000_0000h.

Bits	Description
31:16	**IOLimit[31:16]**. Read-write. Upper part of the limit address. Lower part is defined in D[8:4]F0x1C.
15:0	**IOBase[31:16]**. Read-write. Upper part of the base address. Lower part is defined in D[8:4]F0x1C.

D[8:4]F0x34 Capabilities Pointer

Reset: 0000_0050h.

Bits	Description
31:8	Reserved.
7:0	**CapPtr: capabilities pointer**. Read-only. Pointer to PM capability.

D[8:4]F0x3C Bridge Control

Reset: 0000_00FFh.

Bits	Description
31:24	Reserved.
23	**FastB2BCap: Fast back-to-back capability**. Read-only.
22	**SecondaryBusReset: Secondary bus reset**. Read-write. Setting this bit triggers a hot reset on the corresponding PCI Express Port.
21	**MasterAbortMode: Master abort mode**. Read-only.
20	**Vga16En: VGA IO 16 bit decoding enable**. Read-write. 1= Address bits 15:10 for VGA IO cycles are decoded. 0=Address bits 15:10 for VGA IO cycles are ignored.
19	**VgaEn: VGA enable**. Read-write. Affects the response by the bridge to compatible VGA addresses. When it is set, the bridge decodes and forwards the following accesses on the primary interface to the secondary interface: Memory accesses in the range of A_0000h to B_FFFFh and IO address where address bits 9:0 are in the ranges of 3B0h to 3BBh or 3C0h to 3DFh. For IO cycles the decoding of address bits 15:10 depends on Vga16En.
18	**IsaEn: ISA enable**. Read-write.
17	**SerrEn: SERR enable**. Read-write.
16	**ParityResponseEn: Parity response enable**. Read-write. Controls the bridge's response to poisoned TLPs on its secondary interface. 1=The bridge takes its normal action when a poisoned TLP is received. 0=The bridge ignores any poisoned TLPs that it receives and continues normal operation.
15:11	Reserved.
10:8	**IntPin: interrupt pin**. IF (D0F0xE4_x0101_0010[HwInitWrLock]==1) THEN Read-only. ELSE Read-write. ENDIF.
7:0	**IntLine: Interrupt line**. Read-write.

D[8:4]F0x50 Power Management Capability

Reset: C803_5801h.

Bits	Description
31:27	**PmeSupport**. Read-only.
26	**D2Support: D2 support**. Read-only. D2 is not supported
25	**D1Support: D1 support**. Read-only. D1 is not supported
24:22	**AuxCurrent: auxiliary current**. IF (D0F0xE4_x0101_0010[HwInitWrLock]==1) THEN Read-only. ELSE Read-write. ENDIF. Auxiliary current is not supported.
21	**DevSpecificInit: device specific initialization**. Read-only. This field is hardwired to 0 to indicate that there is no device specific initialization necessary.
20	Reserved.
19	**PmeClock**. Read-only. 0=Indicate that PCI clock is not needed to generate PME messages.
18:16	**Version: version**. Read-only.
15:8	**NextPtr: next pointer**. Read-only. 58h=The address of the next capability structure, or zero if this the end of the linked list of capability structures.
7:0	**CapID: capability ID**. Read-only. 01h=PCI power management data structure.

D[8:4]F0x54 Power Management Control and Status

Bits	Description
31:24	**PmeData**. Read-only. Reset: 0.
23	**BusPwrEn**. Read-only. Reset: 0.
22	**B2B3Support**. Read-only. Reset: 0. B states are not supported.
21:16	Reserved.
15	**PmeStatus: PME status**. Read; set-by-hardware; write-1-to-clear. Reset: 0. This bit is set when the root port would issue a PME message (independent of the state of the PmeEn bit). Once set, this bit remains set until it is reset by writing a 1 to this bit location. Writing a 0 has no effect.
14:13	**DataScale: data scale**. Read-only. Reset: 0.
12:9	**DataSelect: data select**. Read-only. Reset: 0.
8	**PmeEn: PME# enable**. Read-write. Reset: 0.
7:4	Reserved.
3	**NoSoftReset: no soft reset**. Read-only. Reset: 0. Software is required to re-initialize the function when returning from $D3_{hot}$.
2	Reserved.
1:0	**PowerState: power state**. Read-write. Reset: 0. This 2-bit field is used both to determine the current power state of the root port and to set the root port into a new power state. Bits Definition Bits Definition 00b D0 10b Reserved 01b Reserved 11b D3

D[8:4]F0x58 PCI Express Capability

Reset: 0042_B010h.

Bits	Description
31:30	Reserved.
29:25	**IntMessageNum: interrupt message number**. Read-only. This register indicates which MSI vector is used for the interrupt message.
24	**SlotImplemented: Slot implemented**. IF (D0F0xE4_x0101_0010[HwInitWrLock]==1) THEN Read-only. ELSE Read-write. ENDIF. 1=The IO Link associated with this port is connected to a slot.
23:20	**DeviceType: device type**. Read-only. 4h=Root complex.
19:16	**Version**. Read-only. 2h=GEN 2 compliant.
15:8	**NextPtr: next pointer**. Read-only. The address of the next capability structure, or zero if this the end of the linked list of capability structures.
7:0	**CapID: capability ID**. Read-only. 10h=PCIe® Capability structure.

D[8:4]F0x5C Device Capability

Reset: 0000_8020h.

Bits	Description
31:29	Reserved.
28	**FlrCapable: function level reset capability**. Read-only.
27:26	**CapturedSlotPowerScale: captured slot power limit scale**. Read-only.
25:18	**CapturedSlotPowerLimit: captured slot power limit value**. Read-only.
17:16	Reserved.
15	**RoleBasedErrReporting: role-based error reporting**. Read-only.
14:12	Reserved.
11:9	**L1AcceptableLatency: endpoint L1 Acceptable Latency**. Read-only.
8:6	**L0SAcceptableLatency: endpoint L0s Acceptable Latency**. Read-only.
5	**ExtendedTag: extended tag support**. Read-only. 1=8 bit tag supported. 0=5 bit tag supported.
4:3	**PhantomFunc: phantom function support**. Read-only. 0=No phantom functions supported.
2:0	**MaxPayloadSupport: maximum supported payload size**. Read-only. 000b=128 bytes max payload size.

D[8:4]F0x60 Device Control and Status

Reset: 0000_2810h.

Bits	Description
31:22	Reserved.
21	**TransactionsPending: transactions pending**. Read-only. 0=No internally generated non-posted transactions pending.

20	**AuxPwr: auxiliary power**. IF (D0F0xE4_x0101_0010[HwInitWrLock]==1) THEN Read-only. ELSE Read-write. ENDIF.
19	**UsrDetected: unsupported request detected**. Read; set-by-hardware; write-1-to-clear. 1=The port received an unsupported request. Errors are logged in this register even if error reporting is disabled.
18	**FatalErr: fatal error detected**. Read; set-by-hardware; write-1-to-clear. 1=The port detected a fatal error. Errors are logged in this register even if error reporting is disabled.
17	**NonFatalErr: non-fatal error detected**. Read; set-by-hardware; write-1-to-clear. T1=The port detected a non-fatal error. Errors are logged in this register even if error reporting is disabled.
16	**CorrErr: correctable error detected**. Read; set-by-hardware; write-1-to-clear. 1=The port detected a correctable error. Errors are logged in this register even if error reporting is disabled.
15	**BridgeCfgRetryEn: bridge configuration retry enable**. Read-only.
14:12	**MaxRequestSize: maximum request size**. Read-write.
11	**NoSnoopEnable: enable no snoop**. Read-write. 1=The port is permitted to set the No Snoop bit in the Requester Attributes of transactions it initiates that do not require hardware enforced cache coherency.
10	**AuxPowerPmEn: auxiliary power PM enable**. Read-only.
9	**PhantomFuncEn: phantom functions enable**. Read-only.
8	**ExtendedTagEn: extended tag enable**. Read-write. 1=8-bit tags generation enabled. 0=5-bit tags are used.
7:5	**MaxPayloadSize: maximum supported payload size**. Read-write. Bits Definition Bits Definition 0h 128 bytes 4h 2048 bytes 1h 256 bytes 5h 4096 bytes 2h 512 bytes 6h Reserved 3h 1024 bytes 7h Reserved
4	**RelaxedOrdEn: relaxed ordering enable**. Read-write. 1=The root port is permitted to set the relaxed ordering bit in the attributes field of transactions it initiates that do not require strong write ordering.
3	**UsrReportEn: unsupported request reporting enable**. Read-write. 1=Reporting of unsupported requests enabled.
2	**FatalErrEn: fatal error reporting enable**. Read-write. 1=Enable sending ERR_FATAL messages.
1	**NonFatalErrEn: non-fatal error reporting enable**. Read-write. 1=Enable sending ERR_NONFATAL messages.
0	**CorrErrEn: correctable error reporting enable**. Read-write. 1=Enable sending ERR_CORR messages.

D[8:4]F0x64 IO Link Capability

Read-only.

Table 51: Reset mapping for D[8:4]F0x64

Register	Reset
D4F0x64	F710_0C12h.
D5F0x64	F710_0C12h.

Table 51: Reset mapping for D[8:4]F0x64

D6F0x64	F710_0C12h.
D7F0x64	F710_0C12h.
D8F0x64	F710_0C42h.

Bits	Description
31:24	**PortNumber: port number**. This field indicates the port number for the given IO link.
23:22	Reserved.
21	**LinkBWNotificationCap: link bandwidth notification capability**.
20	**DlActiveReportingCapable: data link layer active reporting capability**.
19	Reserved.
18	**ClockPowerManagement: clock power management**. 0=Indicates that the reference clock must not be removed while in L1 or L2/L3 ready link states.
17:15	**L1ExitLatency: L1 exit latency**. 010b=Indicate an exit latency between 2 us and 4 us.
14:12	**L0sExitLatency: L0s exit latency**. 001b=Indicates an exit latency between 64 ns and 128 ns.
11:10	**PMSupport: active state power management support**.11b=Indicates support of L0s and L1.
9:4	**LinkWidth: maximum link width**. Bits — Definition 00h — Reserved 01h — 1 lane 02h — 2 lanes 03h — Reserved 04h — 4 lanes 07h-05h — Reserved 08h — 8 lanes 0Bh-09h — Reserved 0Ch — 12 lanes 0Fh-0Dh — Reserved 10h — 16 lanes 3Fh-11h — Reserved
3:0	**LinkSpeed: link speed**. Bits — Definition 0h — Reserved 1h — 2.5 Gb/s 2h — 5.0 Gb/s Fh-3h — Reserved

D[8:4]F0x68 IO Link Control and Status

Reset: 1100_0000h.

Bits	Description
31	**LinkAutonomousBWStatus: link autonomous bandwidth status**. IF (D[8:4]F0x64[LinkBWNotificationCap]==0) THEN Read-only. ELSE Read-write. ENDIF.

30	**LinkBWManagementStatus: link bandwidth management status**. IF (D[8:4]F0x64[LinkBWNotificationCap]==0) THEN Read-only. ELSE Read-write. ENDIF.
29	**DlActive: data link layer link active**. Read-only. This bit indicates the status of the data link control and management state machine. 1=DL_Active state. 0=All other states.
28	**SlotClockCfg: slot clock configuration**. Read-only. 1=The root port uses the same clock that the platform provides.
27	**LinkTraining: link training**. Read-only. This read-only bit indicates that the physical layer link training state machine is in the configuration or recovery state, or that 1b was written to the RetrainLink bit but link training has not yet begun. Hardware clears this bit when the link training state machine exits the configuration/recovery state.
26	Reserved.
25:20	**NegotiatedLinkWidth: negotiated link width**. Read-only. This field indicates the negotiated width of the given PCI Express link. Bits — Definition 01h — 1 lane 02h — 2 lanes 04h — 4 lanes 08h — 8 lanes 0Ch — 12 lanes 10h — 16 lanes
19:16	**LinkSpeed: link speed**. Read-only. Bits — Definition 0001b — 2.5 Gb/s 0010b — 5 Gb/s
15:12	Reserved.
11	**LinkAutonomousBWIntEn: link autonomous bandwidth interrupt enable**. Read-only. 1=Enables the generation of an interrupt to indicate that the Link AutonomousBWStatus bit has been set.
10	**LinkBWManagementEn: link bandwidth management interrupt enable**. Read-only. 1=Enables the generation of an interrupt to indicate that the LinkBWManagementStatus has been set.
9	**HWAutonomousWidthDisable: hardware autonomous width disable**. Read-write. 1=Disables hardware from changing the link width for reasons other than attempting to correct unreliable link operation by reducing link width.
8	**ClockPowerManagementEn: clock power management enable**. Read-only.
7	**ExtendedSync: extended sync**. Read-write. 1=Forces the transmission of additional ordered sets when exiting the L0s state and when in the recovery state.
6	**CommonClockCfg: common clock configuration**. Read-write. 1=Indicates that the root port and the component at the opposite end of this IO link are operating with a distributed common reference clock. 0=Indicates that the root port and the component at the opposite end of this IO Link are operating with asynchronous reference clock.
5	**RetrainLink: retrain link**. RAZ; write-1-only; cleared-when-done. 1=Initiate link retraining. Reads of this bit always return 0.
4	**LinkDis: link disable**. Read-write. 1=Disable link. Writes to this bit are immediately reflected in the value read from the bit, regardless of actual link state.
3	**ReadCplBoundary: read completion boundary**. Read-only. 0=64 byte read completion boundary.

2	Reserved.
1:0	**PmControl: active state power management enable**. Read-write. This field controls the level of ASPM supported on the given IO link. Bits Definition Bits Definition 00b Disabled 10b L1 Entry Enabled 01b L0s Entry Enabled 11b L0s and L1 Entry Enabled

D[8:4]F0x6C Slot Capability

Reset: 0004_0000h.

Bits	Description
31:19	**PhysicalSlotNumber: physical slot number**. IF (D0F0xE4_x0101_0010[HwInitWrLock]==1) THEN Read-only. ELSE Read-write. ENDIF. This field indicates the physical slot number attached to this port. This field is set to a value that assigns a slot number that is unique within the chassis, regardless of the form factor associated with the slot. This field must be initialized to 0 for ports connected to devices that are on the system board.
18	**NoCmdCplSupport: no command completed support**. IF (D0F0xE4_x0101_0010[HwInitWrLock]==1) THEN Read-only. ELSE Read-write. ENDIF. 1 =Indicates that this slot does not generate software notification when an issued command is completed by the hot-plug controller.
17	**ElecMechIlPresent: electromechanical interlock present**. IF (D0F0xE4_x0101_0010[HwInitWrLock]==1) THEN Read-only. ELSE Read-write. ENDIF. 0=Indicates that a electromechanical interlock is not implemented for this slot.
16:15	**SlotPwrLimitScale: slot power limit scale**. IF (D0F0xE4_x0101_0010[HwInitWrLock]==1) THEN Read-only. ELSE Read-write. ENDIF. Specifies the scale used for the SlotPwrLimitValue. Range of Values: Bits Definition Bits Definition 00b 1.0 10b 0.01 01b 0.1 11b 0.001
14:7	**SlotPwrLimitValue: slot power limit value**. IF (D0F0xE4_x0101_0010[HwInitWrLock]==1) THEN Read-only. ELSE Read-write. ENDIF. In combination with the SlotPwrLimitScale value, specifies the upper limit on power supplied by slot. Power limit (in Watts) calculated by multiplying the value in this field by the value in the SlotPwrLimitScale field.
6	**HotplugCapable: hot-plug capability**. IF (D0F0xE4_x0101_0010[HwInitWrLock]==1) THEN Read-only. ELSE Read-write. ENDIF. 1=Indicates that this slot is capable of supporting hot-plug operations.
5	**HotplugSurprise: hot-plug surprise**. IF (D0F0xE4_x0101_0010[HwInitWrLock]==1) THEN Read-only. ELSE Read-write. ENDIF. 1=Indicates that an adapter present in this slot might be removed from the system without any prior notification.
4	**PwrIndicatorPresent: power indicator present**. IF (D0F0xE4_x0101_0010[HwInitWrLock]==1) THEN Read-only. ELSE Read-write. ENDIF. 0=Indicates that a power indicator is not implemented for this slot.
3	**AttnIndicatorPresent: attention indicator present**. IF (D0F0xE4_x0101_0010[HwInitWrLock]==1) THEN Read-only. ELSE Read-write. ENDIF. 0=Indicates that a attention indicator is not implemented for this slot.

2	**MrlSensorPresent: manual retention latch sensor present**. IF (D0F0xE4_x0101_0010[HwInitWrLock]==1) THEN Read-only. ELSE Read-write. ENDIF. 0=Indicates that a manual retention latch sensor is not implemented for this slot.
1	**PwrControllerPresent: power controller present**. IF (D0F0xE4_x0101_0010[HwInitWrLock]==1) THEN Read-only. ELSE Read-write. ENDIF. 0=A power controller is not implemented for this slot.
0	**AttnButtonPresent: attention button present**. IF (D0F0xE4_x0101_0010[HwInitWrLock]==1) THEN Read-only. ELSE Read-write. ENDIF. 0=An attention button is not implemented for this slot.

D[8:4]F0x70 Slot Control and Status

IF (D[8:4]F0x58[SlotImplemented]==0) THEN Reset: 0040_0000h. ELSE Reset: 0000_0000h. ENDIF.

Bits	Description
31:25	Reserved.
24	**DlStateChanged: data link layer state change**. Read; set-by-hardware; write-1-to-clear. This bit is set when the value reported in the D[8:4]F0x60[DlActive] is changed. In response to a data link layer state changed event, software must read D[8:4]F0x60[DlActive] to determine if the link is active before initiating configuration cycles to the hot plugged device.
23	**ElecMechIlSts: electromechanical interlock status**. Read-only.
22	**PresenceDetectState: presence detect state**. Read-only. This bit indicates the presence of an adapter in the slot based on the physical layer in-band presence detect mechanism. The in-band presence detect mechanism requires that power be applied to an adapter for its presence to be detected. 0=Slot empty. 1=Card present in slot. For root ports not connected to slots (D[8:4]F0x58[SlotImplemented]==0b), this bit returns always 1.
21	**MrlSensorState**. Read-only. The current state of the manual retention latch sensor.
20	**CmdCpl: command completed**. Read; set-by-hardware; write-1-to-clear. 1=The hot-plug controller has completed an issued command.
19	**PresenceDetectChanged: presence detect changes**. Read; set-by-hardware; write-1-to-clear. 1=D[8:4]F0x70[PresenceDetectState] changed.
18	**MrlSensorChanged**. Read; set-by-hardware; write-1-to-clear. 1=The manual retention latch sensor changed state.
17	**PwrFaultDetected**. Read; set-by-hardware; write-1-to-clear. 1=The power controller detected a power fault at this slot.
16	**AttnButtonPressed: attention button pressed**. Read; set-by-hardware; write-1-to-clear. 1=The attention button has been pressed.
15:13	Reserved.
12	**DlStateChangedEn: data link layer state changed enable**. Read-write. 1=Enables software notification when D[8:4]F0x60[DlActive] is changed.
11	**ElecMechIlCntl: electromechanical interlock control**. RAZ.
10	**PwrControllerCntl: power controller control**. IF (D[8:4]F0x6C[PwrControllerPresent]==1) THEN Read-write; updated-by-hardware. ELSE Read-only; updated-by-hardware. ENDIF. Writes to this field set the power state of the slot to predefined encodings. Reads from this register return the current state of the power applied to the slot.

9:8	**PwrIndicatorCntl: power indicator control**. IF (D[8:4]F0x6C[PwrIndicatorPresent]==1) THEN Read-write; updated-by-hardware. ELSE Read-only; updated-by-hardware. ENDIF. Writes to this field set the power indicator. Reads from this register return the current state of the power indicator.
7:6	**AttnIndicatorControl: attention indicator control**. IF (D[8:4]F0x6C[AttnIndicatorPresent]==1) THEN Read-write; updated-by-hardware. ELSE Read-only; updated-by-hardware. ENDIF. Writes to this field set the attention indicator. Reads from this register return the current state of the attention indicator.
5	**HotplugIntrEn: hot-plug interrupt enable**. IF (D[8:4]F0x6C[HotplugCapable]==1) THEN Read-write. ELSE Read-only. ENDIF. 1=Enables generation of a hot-plug interrupt on enabled hot-plug events.
4	**CmdCplIntrEn: command complete interrupt enable**. IF (D[8:4]F0x6C[NoCmdCplSupport]==1) THEN Read-only. ELSE Read-write. ENDIF. 1=Enables generation of a hot-plug interrupt when a command is completed by the hot-plug controller.
3	**PresenceDetectChangedEn: presence detect changed enable**. IF (D[8:4]F0x6C[HotplugCapable]==1) THEN Read-write. ELSE Read-only. ENDIF. 1=Enables generation of a hot-plug interrupt or wake-up on a presence detect changed event.
2	**MrlSensorChangedEn: manual retention latch sensor changed enable**. IF (D[8:4]F0x6C[MrlSensorPresent]==1) THEN Read-write. ELSE Read-only. ENDIF. 1=Enables software notification on a manual retention latch sensor change event.
1	**PwrFaultDetectedEn: power fault detected enable**. Read-only. 1=Enables software notification on a power fault event.
0	**AttnButtonPressedEn: attention button pressed enable**. IF (D[8:4]F0x6C[AttnButtonPresent]==1) THEN Read-write. ELSE Read-only. ENDIF. 1=Enables generation of a hot-plug interrupt or wake-up on an attention button pressed event.

D[8:4]F0x74 Root Complex Capability and Control

Reset: 0001_0000h.

Bits	Description
31:17	Reserved.
16	**CrsSoftVisibility: CRS software visibility**. Read-only. 1=Indicates that the root port supports returning configuration request retry status (CRS) completion status to software.
15:5	Reserved.
4	**CrsSoftVisibilityEn: CRS software visibility enable**. Read-write. 1=Enables the root port returning configuration request retry status (CRS) completion status to software.
3	**PmIntEn: PME interrupt enable**. Read-write. 1=Enables interrupt generation upon receipt of a PME message as reflected D[8:4]F0x78[PmeStatus]. A PME interrupt is also generated if D[8:4]F0x78[PmeStatus]=1 and this bit is set by software.
2	**SerrOnFatalErrEn: system error on fatal error enable**. Read-write. 1=Indicates that a system error should be generated if a fatal error (ERR_FATAL) is reported by any of the devices in the hierarchy associated with this root port, or by the root port itself.

1	**SerrOnNonFatalErrEn: system error on non-fatal error enable**. Read-write. 1=Indicates that a system error should be generated if a non-fatal error (ERR_NONFATAL) is reported by any of the devices in the hierarchy associated with this root port, or by the root port itself.
0	**SerrOnCorrErrEn: system error on correctable error enable**. Read-write. 1=Indicates that a system error should be generated if a correctable error (ERR_COR) is reported by any of the devices in the hierarchy associated with this root port, or by the root port itself.

D[8:4]F0x78 Root Complex Status

Reset: 0000_0000h.

Bits	Description
31:18	Reserved.
17	**PmePending: PME pending**. Read-only; updated-by-hardware. This bit indicates that another PME is pending when PmeStatus is set. When PmeStatus is cleared by software; the PME is delivered by hardware by setting the PmeStatus bit again and updating the requestor ID appropriately. PmePending is cleared by hardware if no more PMEs are pending.
16	**PmeStatus: PME status**. Read; set-by-hardware; write-1-to-clear. This bit indicates that PME was asserted by the requestor ID indicated in the PmeRequestorID field. Subsequent PMEs are kept pending until PmeStatus is cleared by writing a 1.
15:0	**PmeRequestorId: PME requestor ID**. Read-only; updated-by-hardware. This field indicates the PCI requestor ID of the last PME requestor.

D[8:4]F0x7C Device Capability 2

Reset: 0000_001Fh.

Bits	Description
31:6	Reserved.
5	**AriForwardingSupported**. Read-only. Program D0F0xE4_x0130_0000[StrapBifAriEn]=1 and D0F0xE4_x0101_00C1[StrapGen2Compliance]=1 to enable this feature.
4	**CplTimeoutDisSup: completion timeout disable supported**. Read-only.
3:0	**CplTimeoutRangeSup: completion timeout range supported**. Read-only. Fh=Completion timeout range is 64 s to 50 us.

D[8:4]F0x80 Device Control and Status 2

Reset: 0000_0000h.

Bits	Description
31:6	Reserved.
5	**AriForwardingEn**. IF (D[8:4]F0x7C[AriForwardingSupported]==1) THEN Read-write. ELSE Read-only. ENDIF.

4	**CplTimeoutDis: completion timeout disable**. Read-write. 1=Completion timeout disabled.
3:0	**CplTimeoutValue: completion timeout value**. Read-write. BIOS: 6h. Must be less than the WDT timeout specified by D18F3x44[WDTBaseSel, WDTCntSel]. Bits / Timeout Range: 0h: 50 us to 50 ms; 1h: 50 us to 100 us; 2h: 1 ms to 10 ms; 4h-3h: Reserved; 5h: 16 ms to 55 ms; 6h: 65 ms to 210 ms; 8h-7h: Reserved; 9h: 260 ms to 900 ms; Ah: 1 s to 3.5 s; Ch-Bh: Reserved; Dh: 4 s to 13 s; Eh: 4 s to 64 s; Fh: Reserved

D[8:4]F0x84 IO Link Capability 2

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D[8:4]F0x88 IO Link Control and Status 2

Bits	Description
31:17	Reserved.
16	**CurDeemphasisLevel: current de-emphasis level**. Read-only; updated-by-hardware. Reset: D[8:4]F0xE4_xA4[LcGen2EnStrap]. 1=-3.5 dB. 0=-6 dB
15:13	Reserved.
12	**ComplianceDeemphasis: compliance de-emphasis**. Read-write. Cold reset: 0. This bit defines the compliance de-emphasis level when EnterCompliance is set. Software should leave this field in its default state. 1=3.5 dB. 0=-6 dB
11	**ComplianceSOS: compliance SOS**. Read-write. Cold reset: 0. 1=The device transmits skip ordered sets in between the modified compliance pattern.
10	**EnterModCompliance: enter modified compliance**. Read-write. Cold reset: 0. 1=The device transmits modified compliance pattern. Software should leave this field in its default state.
9:7	**XmitMargin: transmit margin**. Read-write. Cold reset: 0. This field controls the voltage level (without de-emphasis) at the transmitter pins. Software should leave this field in its default state.
6	**SelectableDeemphasis: selectable de-emphasis**. Read-only. Reset: D[8:4]F0xE4_xA4[LcGen2EnStrap]. 1=Selectable de-emphasis is supported. 0=Selectable de-emphasis is not supported.
5	**HwAutonomousSpeedDisable: hardware autonomous speed disable**. Read-write. Cold reset: 0. 1=Support for hardware changing the link speed for device specific reasons disabled.

4	**EnterCompliance: enter compliance**. Read-write. Cold reset: 0. 1=Force the link to enter the compliance mode.
3:0	**TargetLinkSpeed: target link speed**. Read-write; updated-by-hardware. Cold reset: {00b, D[8:4]F0xE4_xA4[LcGen2EnStrap], ~D[8:4]F0xE4_xA4[LcGen2EnStrap]}. This field defines the upper limit of the link operational speed. When D[8:4]F0xE4_xA4[LcGen2EnStrap]=1, this field can only be programmed to 1h or 2h. When D[8:4]F0xE4_xA4[LcGen2EnStrap]=0, this field can only be programmed to 1h. If this field has not been written since reset, hardware will change this field to {00b, D[8:4]F0xE4_xA4[LcGen2EnStrap], ~D[8:4]F0xE4_xA4[LcGen2EnStrap]} when D[8:4]F0xE4_xA4[LcGen2EnStrap] changes. Bits — Definition 0h — Reserved 1h — 2.5 GT/s 2h — 5.0 GT/s Fh-3h — Reserved

D[8:4]F0x8C Slot Capability 2

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D[8:4]F0x90 Slot Control and Status 2

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D[8:4]F0xA0 MSI Capability

Reset: 0000_B005h.

Bits	Description
31:24	Reserved.
23	**Msi64bit: MSI 64 bit capability**. Read-only. Value: D0F0x64_x46[Msi64bitEn]. 1=The device is capable of sending 64-bit MSI messages. 0=The device is not capable of sending a 64-bit message address.
22:20	**MsiMultiEn: MSI multiple message enable**. Read-write. Reset: 000b. Software writes to this field to indicate the number of allocated vectors (equal to or less than the number of requested vectors). When MSI is enabled, a function is allocated at least 1 vector.
19:17	**MsiMultiCap: MSI multiple message capability**. Read-only. Reset: 000b. 000b=The device is requesting one vector.
16	**MsiEn: MSI enable**. Read-write. Reset: 0. 1=MSI generation is enabled and INTx generation is disabled. 0=MSI generation disabled and INTx generation is enabled.

15:8	**NextPtr: next pointer**. Read-only. Reset: B0h. The address of the next capability structure, or zero if this the end of the linked list of capability structures.
7:0	**CapID: capability ID**. Read-only. Reset: 05h. 05h=MSI capability structure.

D[8:4]F0xA4 MSI Message Address Low

Reset: 0000_0000h.

Bits	Description
31:2	**MsiMsgAddrLo: MSI message address**. IF (D0F0xE4_x0101_00B0[StrapF0MsiEn]==1) THEN Read-write. ELSE Read-only. ENDIF. This register specifies the doubleword aligned address for the MSI memory write transaction.
1:0	Reserved.

IF (D0F0x64_x46[Msi64bitEn]==1) THEN

D[8:4]F0xA8 MSI Message Address High

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**MsiMsgAddrHi: MSI message address**. IF (D0F0xE4_x0101_00B0[StrapF0MsiEn]==1) THEN Read-write. ELSE Read-only. ENDIF. This register specifies the upper 8 bits of the MSI address in 64-bit MSI mode.

ENDIF.

IF (D0F0x64_x46[Msi64bitEn]==0) THEN

D[8:4]F0xA8 MSI Message Data

ELSE

D[8:4]F0xAC MSI Message Data

ENDIF.
Reset: 0000_0000h.

Bits	Description
31:16	Reserved.
15:0	**MsiData: MSI message data**. IF (D0F0xE4_x0101_00B0[StrapF0MsiEn]==1) THEN Read-write. ELSE Read-only. ENDIF. This register specifies the lower 16 bits of data for the MSI memory write transaction. The upper 16 bits are always 0.

D[8:4]F0xB0 Subsystem and Subvendor Capability ID

Reset: 0000_B80Dh.

Bits	Description
31:16	Reserved.

15:8	**NextPtr: next pointer**. Read-only. The address of the next capability structure, or zero if this the end of the linked list of capability structures.
7:0	**CapID: capability ID**. Read-only.

D[8:4]F0xB4 Subsystem and Subvendor ID

Bits	Description
31:16	**SubsystemID**. Value: D0F0xE4_x0130_8002[SubsystemID].
15:0	**SubsystemVendorID**. Value: D0F0xE4_x0130_8002[SubsystemVendorID].

D[8:4]F0xB8 MSI Capability Mapping

Reset: A803_0008h.

Bits	Description
31:27	**CapType: capability type**. Read-only.
26:18	Reserved.
17	**FixD**. Read-only.
16	**En**. Read-only.
15:8	**NextPtr: next pointer**. Read-only. The address of the next capability structure, or zero if this the end of the linked list of capability structures.
7:0	**CapID: capability ID**. Read-only.

D[8:4]F0xBC MSI Mapping Address Low

Bits	Description
31:20	**MsiMapAddrLo**. Read-only. Reset: 0. Lower 32 bits of the MSI address.
19:0	Reserved.

D[8:4]F0xC0 MSI Mapping Address High

Bits	Description
31:0	**MsiMapAddrHi**. Read-only. Reset: 0. Upper 32 bits of the MSI address.

D[8:4]F0xE0 Root Port Index

Reset: 0000_0000h.

The index/data pair registers D[8:4]F0xE0 and D[8:4]F0xE4 is used to access the registers D[8:4]F0xE4_x[FF:00]. To read or write to one of these registers, the address is written first into the address register D[8:4]F0xE0 and then the data are read or written by read or write the data register D[8:4]F0xE4.

Bits	Description

31:8	Reserved.
7:0	**PcieIndex**. Read-write.

D[8:4]F0xE4 Root Port Data

Reset: FFFF_FFFFh. See D[8:4]F0xE0.

Bits	Description
31:0	**PcieData**. Read-write.

D[8:4]F0xE4_x50 Root Port Lane Status

Reset: 0000_0000h.

Bits	Description
31:7	Reserved.
6:1	**PhyLinkWidth: port link width**. Read-only. Bits / Definition / Bits / Definition 00_0000b disabled 00_1000b x8 00_0001b x1 01_0000b Reserved 00_0010b x2 10_0000b Reserved 00_0100b x4
0	**PortLaneReversal: port lane reversal**. Read-only. 1=Port lanes order is reversed.

D[8:4]F0xE4_x6A Root Port Error Control

Reset: 0000_0500h.

Bits	Description
31:1	Reserved.
0	**ErrReportingDis: advanced error reporting disable**. Read-write. BIOS: 0. 1=Error reporting disabled. 0=Error reporting enabled.

D[8:4]F0xE4_x70 Root Port Receiver Control

Reset: 0000_43F7h.

Bits	Description
31:20	Reserved.
19	**RxRcbCplTimeoutMode: RCB completion timeout mode**. Read-write. BIOS: 1.
18:16	**RxRcbCplTimeout: RCB completion timeout**. Read-write. Bits / Definition / Bits / Definition 000b Disabled 100b 50 ms 001b 50 us 101b 100 ms 010b 10 ms 110b 500 ms 011b 25 ms 111b 1 ms
15:0	Reserved.

D[8:4]F0xE4_xA4 LC Link Speed Control

Reset: 1202_8041h.

Bits	Description
31:30	Reserved.
29	**LcMultUpstreamAutoSpdChngEn: enable multiple automatic speed changes**. Read-write. 1=Enable multiple automatic speed changes.
28:25	Reserved.
24	**LcOtherSideSupportsGen2: downstream link supports Gen2**. Read-only. 1=The downstream link currently supports Gen2.
23:19	Reserved.
18	**LcGoToRecovery: go to recovery**. Read-write. 1=Force link in the L0 state to transition to the recovery state.
17:11	Reserved.
10	**LcSpeedChangeAttemptFailed: speed change attempt failed**. Read-only; updated-by-hardware. 1=LcSpeedChangeAttemptsAllowed has been reached.
9:8	**LcSpeedChangeAttemptsAllowed: speed change attempts allowed**. Read-write. Determines the number of speed change attempts that are allowed.
7	**LcInitiateLinkSpeedChange: initiate link speed change**. Read-write; cleared-when-done. 1=Initiate link speed negotiation.
6:5	Reserved.
4	**LcForceDisSwSpeedChange: force disable software speed changes**. Read-write. 1=Force the PCIe core to disable speed changes initiated by private registers.
3:1	Reserved.
0	**LcGen2EnStrap: Gen2 PCIe support enable**. Read-write. 1=Gen2 PCIe support enabled. 0=Gen2 PCIe support disabled.

D[8:4]F0xE4_xB5 LC Control 3

Reset: 0050_5028h.

Bits	Description
31:4	Reserved.
3	**LcRcvdDeemphasis: received de-emphasis**. Read-only; updated-by-hardware. De-emphasis advertised by the downstream device. 1=3.5 dB. 0=6 dB.
2:1	**LcSelectDeemphasisCntl: de-emphasis control**. Read-write. Specifies the de-emphasis used by the transmitter. Bits Definition 00b Use de-emphasis from LcSelectDeemphasis. 01b Use de-emphasis advertised by the downstream device. 10b 6 dB 11b 3.5 dB
0	**LcSelectDeemphasis: downstream de-emphasis**. Read-write. Specifies the downstream de-emphasis. 1=3.5 dB. 0=6 dB.

D[8:4]F0xE4_xC0 LC Strap Override

Bits	Description
31:16	Reserved.
15	**StrapAutoRcSpeedNegotiationDis: autonomous speed negotiation disable strap override**. Read-write. Reset: 1. SBIOS: 1.
14	Reserved.
13	**StrapForceCompliance: force compliance strap override**. Read-write. Reset: 0b.
12:0	Reserved.

D[8:4]F0xE4_xC1 Root Port Miscellaneous Strap Override

Reset: 0000_0000h.

Bits	Description
31:5	Reserved.
4	**StrapReverseLanes: reverse lanes strap override**. Read-write.
3:0	Reserved.

D[8:4]F0x100 Vendor Specific Enhanced Capability

Bits	Description
31:20	**NextPtr: next pointer**. Read-only. Reset: 000h. The address of the next capability structure, or zero if this the end of the linked list of capability structures.
19:16	**CapVer: capability version**. Read-only. Reset: 1h.
15:0	**CapID: capability ID**. Read-only. Reset: 000Bh.

D[8:4]F0x104 Vendor Specific Header

Reset: 0101_0001h.

Bits	Description
31:20	**VsecLen: vendor specific enhanced capability structure length**. Read-only. Defined the number of bytes of the entire vendor specific enhanced capability structure including the header.
19:16	**VsecRev: vendor specific enhanced capability version**. Read-only.
15:0	**VsecID: vendor specific enhanced capability ID**. Read-only.

D[8:4]F0x108 Vendor Specific 1

Reset: 0000_0000h.

Bits	Description
31:0	**Scratch: scratch**. Read-write. This field does not control any hardware.

D[8:4]F0x10C Vendor Specific 2

Reset: 0000_0000h.

Bits	Description
31:0	**Scratch: scratch**. Read-write. This field does not control any hardware.

D[8:4]F0x150 Advanced Error Reporting Capability

Reset: 0001_0001h.

Bits	Description
31:20	**NextPtr: next pointer**. Read-only. The address of the next capability structure, or zero if this the end of the linked list of capability structures.
19:16	**CapVer: capability version**. Read-only.
15:0	**CapID: capability ID**. Read-only.

D[8:4]F0x154 Uncorrectable Error Status

Cold reset: 0000_0000h.

Bits	Description
31:26	Reserved.
25	**TlpPrefixStatus: TLP prefix blocked status**. Read-only.
24	**AtomicOpEgressBlockedTLPStatus: atomic op egress blocked TLP status**. Read-only.
23	**McBlockedTLPStatus: MC blocked TLP status**. Read-only.
22	**UncorrInteralErrStatus: uncorrectable internal error status**. Read-only.
21	**AcsViolationStatus: access control service status**. Read; set-by-hardware; write-1-to-clear.
20	**UnsuppReqErrStatus: unsupported request error status**. Read; set-by-hardware; write-1-to-clear. The header of the unsupported request is logged.
19	**EcrcErrStatus: end-to-end CRC error status**. Read; set-by-hardware; write-1-to-clear.
18	**MalTlpStatus: malformed TLP status**. Read; set-by-hardware; write-1-to-clear. The header of the malformed TLP is logged.
17	**RcvOvflStatus: receiver overflow status**. Read-only.
16	**UnexpCplStatus: unexpected completion timeout status**. Read; set-by-hardware; write-1-to-clear. The header of the unexpected completion is logged.
15	**CplAbortErrStatus: completer abort error status**. Read; set-by-hardware; write-1-to-clear.
14	**CplTimeoutStatus: completion timeout status**. Read; set-by-hardware; write-1-to-clear.
13	**FcErrStatus: flow control error status**. Read-only.
12	**PsnErrStatus: poisoned TLP status**. Read; set-by-hardware; write-1-to-clear. The header of the poisoned transaction layer packet is logged.
11:6	Reserved.
5	**SurprdnErrStatus: surprise down error status**. Read-only. 0=Detection and reporting of surprise down errors is not supported.

4	**DlpErrStatus: data link protocol error status**. Read; set-by-hardware; write-1-to-clear.
3:0	Reserved.

D[8:4]F0x158 Uncorrectable Error Mask

Cold reset: 0000_0000h.

Bits	Description
31:26	Reserved.
25	**TlpPrefixMask: TLP prefix blocked mask**. Read-only.
24	**AtomicOpEgressBlockedTLPMask: atomic op egress blocked TLP mask**. Read-only.
23	**McBlockedTLPMask: MC blocked TLP mask**. Read-only.
22	**UncorrInteralErrMask: uncorrectable internal error mask**. Read-only.
21	**AcsViolationMask: access control service mask**. Read-only. 1=ACS violation errors are not reported.
20	**UnsuppReqErrMask: unsupported request error mask**. Read-write. 1=Unsupported request errors are not reported.
19	**EcrcErrMask: end-to-end CRC error mask**. IF (D[8:4]F0x168[EcrcCheckEn] == 1) THEN Read-write. ELSE Read-only. ENDIF.
18	**MalTlpMask: malformed TLP mask**. Read-write. 1=Malformed TLP errors are not reported.
17	**RcvOvflMask: receiver overflow mask**. Read-only.
16	**UnexpCplMask: unexpected completion timeout mask**. Read-write. 1=Unexpected completion errors are not reported.
15	**CplAbortErrMask: completer abort error mask**. Read-only.
14	**CplTimeoutMask: completion timeout mask**. Read-write. 1=Completion timeout errors are not reported.
13	**FcErrMask: flow control error mask**. Read-only.
12	**PsnErrMask: poisoned TLP mask**. Read-write. 1=Poisoned TLP errors are not reported.
11:6	Reserved.
5	**SurprdnErrMask: surprise down error mask**. Read-only.
4	**DlpErrMask: data link protocol error mask**. Read-write. 1=Data link protocol errors are not reported.
3:0	Reserved.

D[8:4]F0x15C Uncorrectable Error Severity

Cold reset: 0006_2030h.

Bits	Description
31:26	Reserved.
25	**TlpPrefixSeverity: TLP prefix blocked severity**. Read-only.
24	**AtomicOpEgressBlockedTLPSeverity: atomic op egress blocked TLP severity**. Read-only.
23	**McBlockedTLPSeverity: MC blocked TLP severity**. Read-only.

22	**UncorrInteralErrSeverity: uncorrectable internal error severity**. Read-only.
21	**AcsViolationSeverity: access control service severity**. Read-only. 1=Fatal error. 0=Non-fatal error.
20	**UnsuppReqErrSeverity: unsupported request error severity**. Read-write. 1=Fatal error. 0=Non-fatal error.
19	**EcrcErrSeverity: end-to-end CRC error severity**. IF (D[8:4]F0x168[EcrcCheckEn] == 1) THEN Read-write. ELSE Read-only. ENDIF.
18	**MalTlpSeverity: malformed TLP severity**. Read-write. 1=Fatal error. 0=Non-fatal error.
17	**RcvOvflSeverity: receiver overflow severity**. Read-only.
16	**UnexpCplSeverity: unexpected completion timeout severity**. Read-write. 1=Fatal error. 0=Non-fatal error.
15	**CplAbortErrSeverity: completer abort error severity**. Read-only.
14	**CplTimeoutSeverity: completion timeout severity**. Read-write. 1=Fatal error. 0=Non-fatal error.
13	**FcErrSeverity: flow control error severity**. Read-only.
12	**PsnErrSeverity: poisoned TLP severity**. Read-write. 1=Fatal error. 0=Non-fatal error.
11:6	Reserved.
5	**SurprdnErrSeverity: surprise down error severity**. Read-only.
4	**DlpErrSeverity: data link protocol error severity**. Read-write. 1=Fatal error. 0=Non-fatal error.
3:0	Reserved.

D[8:4]F0x160 Correctable Error Status

Cold reset: 0000_0000h.

Bits	Description
31:14	Reserved.
13	**AdvisoryNonfatalErrStatus: advisory non-fatal error status**. Read; set-by-hardware; write-1-to-clear. 1=A non-fatal unsupported request errors or a non-fatal unexpected completion errors occurred.
12	**ReplayTimerTimeoutStatus: replay timer timeout status**. Read; set-by-hardware; write-1-to-clear.
11:9	Reserved.
8	**ReplayNumRolloverStatus: replay**. Read; set-by-hardware; write-1-to-clear. 1=The same transaction layer packet has been replayed three times and has caused the link to re-train.
7	**BadDllpStatus: bad data link layer packet status**. Read; set-by-hardware; write-1-to-clear. 1=A link CRC error was detected.
6	**BadTlpStatus: bad transaction layer packet status**. Read; set-by-hardware; write-1-to-clear. 1=A bad non-duplicated sequence ID or a link CRC error was detected.
5:1	Reserved.
0	**RcvErrStatus: receiver error status**. Read-only. 1=An 8B10B or disparity error was detected.

D[8:4]F0x164 Correctable Error Mask

Cold reset: 0000_2000h.

Bits	Description
31:14	Reserved.
13	**AdvisoryNonfatalErrMask: advisory non-fatal error mask**. Read-write. 1=Error is not reported.
12	**ReplayTimerTimeoutMask: replay timer timeout mask**. Read-write. 1=Error is not reported.
11:9	Reserved.
8	**ReplayNumRolloverMask: replay**. Read-write. 1=Error is not reported.
7	**BadDllpMask: bad data link layer packet mask**. Read-write. 1=Error is not reported.
6	**BadTlpMask: bad transaction layer packet mask**. Read-write. 1=Error is not reported.
5:1	Reserved.
0	**RcvErrMask: receiver error mask**. Read-only. 1=Error is not reported.

D[8:4]F0x168 Advanced Error Control

Cold reset: 0000_0000h.

Bits	Description
31:9	Reserved.
8	**EcrcCheckEn: data link protocol error severity**. Read-only. 0=Specifies that end-to-end CRC generation is not supported.
7	**EcrcCheckCap: data link protocol error severity**. Read-only. 0=Specifies that end-to-end CRC check is not supported.
6	**EcrcGenEn: end-to-end CRC enable**. Read-only. 0=Specifies that end-to-end CRC generation is not supported.
5	**EcrcGenCap: end-to-end CRC capability**. Read-only. 0=Specifies that end-to-end CRC generation is not supported.
4:0	**FirstErrPtr: first error pointer**. Read-only. The First Error Pointer identifies the bit position of the first error reported in the Uncorrectable Error Status register.

D[8:4]F0x16C Header Log DW0

Cold reset: 0000_0000h.

Bits	Description
31:0	**TlpHdr: transaction layer packet header log**. Read-only. Contains the header for a transaction layer packet corresponding to a detected error. The upper byte represents byte 0 of the header.

D[8:4]F0x170 Header Log DW1

Cold reset: 0000_0000h.

Bits	Description
31:0	**TlpHdr: transaction layer packet header log**. Read-only. Contains the header for a transaction layer packet corresponding to a detected error. The upper byte represents byte 4 of the header.

D[8:4]F0x174 Header Log DW2

Cold reset: 0000_0000h.

Bits	Description
31:0	**TlpHdr: transaction layer packet header log**. Read-only. Contains the header for a transaction layer packet corresponding to a detected error. The upper byte represents byte 8 of the header.

D[8:4]F0x178 Header Log DW3

Cold reset: 0000_0000h.

Bits	Description
31:0	**TlpHdr: transaction layer packet header log**. Read-only. Contains the header for a transaction layer packet corresponding to a detected error. The upper byte represents byte 12 of the header.

D[8:4]F0x17C Root Error Command

Reset: 0000_0000h.

Bits	Description
31:3	Reserved.
2	**FatalErrRepEn: fatal error reporting enable**. Read-write. 1=Enables the generation of an interrupt when a fatal error is reported by any of the devices in the hierarchy associated with this Root Port.
1	**NonfatalErrRepEn: non-fatal error reporting enable**. Read-write. 1=Enables generation of an interrupt when a non-fatal error is reported by any of the devices in the hierarchy associated with this Root Port.
0	**CorrErrRepEn: correctable error reporting enable**. Read-write. 1=Enables generation of an interrupt when a correctable error is reported by any of the devices in the hierarchy associated with this Root Port.

D[8:4]F0x180 Root Error Status

Cold reset: 0000_0000h.

Bits	Description
31:27	**AdvErrIntMsgNum: advanced error interrupt message number**. Read-only.
26:7	Reserved.
6	**NFatalErrMsgRcvd: fatal error message received**. Read; set-by-hardware; write-1-to-clear. Set to 1 when one or more fatal uncorrectable error messages have been received.

5	**NonFatalErrMsgRcvd: non-fatal error message received**. Read; set-by-hardware; write-1-to-clear. Set to 1 when one or more non-fatal uncorrectable error messages have been received.
4	**FirstUncorrFatalRcvd: first uncorrectable fatal error message received**. Read; set-by-hardware; write-1-to-clear. Set to 1 when the first uncorrectable error message received is for a fatal error.
3	**MultErrFatalNonfatalRcvd: ERR_FATAL/NONFATAL message received**. Read; set-by-hardware; write-1-to-clear. Set when either a fatal or a non-fatal error is received and ErrFatalNonfatalRcvd is already set.
2	**ErrFatalNonfatalRcvd: ERR_FATAL/NONFATAL message received**. Read; set-by-hardware; write-1-to-clear. Set when either a fatal or a non-fatal error is received and this bit is not already set.
1	**MultErrCorrRcvd: multiple ERR_COR messages received**. Read; set-by-hardware; write-1-to-clear. Set when a correctable error message is received and ErrCorrRcvd is already set.
0	**ErrCorrRcvd: ERR_COR message received**. Read; set-by-hardware; write-1-to-clear. Set when a correctable error message is received and this bit is not already set.

D[8:4]F0x184 Error Source ID

Cold reset: 0000_0000h.

Bits	Description
31:16	**ErrFatalNonfatalSrcID: ERR_FATAL/ERR_NONFATAL source identification**. Read-only. Loaded with the requestor ID indicated in the received ERR_FATAL or ERR_NONFATAL message when D[8:4]F0x180[ErrFatalNonfatalRcvd] is not already set.
15:0	**ErrCorlSrcID: ERR_COR source identification**. Read-only. Loaded with the requestor ID indicated in the received ERR_COR message when D[8:4]F0x180[ErrCorrRcvd] is not already set.

3.7 Device 18h Function 0 Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space.

D18F0x00 Device/Vendor ID

Reset: 1700_1022h.

Bits	Description
31:16	**DeviceID: device ID**. Read-only.
15:0	**VendorID: vendor ID**. Read-only.

D18F0x04 Status/Command

Reset: 0010_0000h.

Bits	Description
31:16	**Status**. Read-only. Bit[20] is set to indicate the existence of a PCI-defined capability block.
15:0	**Command**. Read-only.

D18F0x08 Class Code/Revision ID

Reset: 0600_0043h.

Bits	Description
31:8	**ClassCode**. Read-only. Provides the host bridge class code as defined in the PCI specification.
7:0	**RevID: revision ID**. Read-only.

D18F0x0C Header Type

Bits	Description
31:0	**HeaderTypeReg**. Value: 0080_0000h. The header type field indicates that there are multiple functions present in this device.

D18F0x34 Capabilities Pointer

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**CapPtr: capabilities pointer**. Read-only. There is no capability block.

D18F0x68 Link Transaction Control

Bits	Description
31:24	Reserved.
23	**InstallStateS**. Read-write. Reset: 0. 1=Forces the default read block (RdBlk) install state to be shared instead of exclusive.
22:21	**DsNpReqLmt: downstream non-posted request limit**. Read-write. Reset: 0. BIOS: 01b. Specifies the maximum number of downstream non-posted requests, issued by core(s) or peer-to-peer, which may be outstanding in the root complex. Bits / Definition / Bits / Definition 00b: no limit. 10b: limited to 4. 01b: limited to 1. 11b: limited to 8.
20	**DisSeqIdReqUID: disable using requester UID for SeqID**. Read-write. Reset: 0. 0=SeqID is assigned the UID of the requesting CPU, such that CPU0=2h and CPU1=3h. 1=SeqID of 0h is used.
19	**ApicExtSpur: APIC extended spurious vector enable**. Read-write. Reset: 0. This enables the extended APIC spurious vector functionality. 0=The lower 4 bits of the spurious vector are read-only 1111b. 1=The lower 4 bits of the spurious vector are writable.
18	**ApicExtId: APIC extended ID enable**. Read-write. Reset: 0. This enables the extended APIC ID functionality. 0=APIC ID is 4 bits. 1=APIC ID is 8 bits.
17	**ApicExtBrdCst: APIC extended broadcast enable**. Read-write. Reset: 0. This enables the extended APIC broadcast functionality. 0=APIC broadcast is 0Fh. 1=APIC broadcast is FFh.

16	**LintEn: local interrupt conversion enable**. Read-write. Reset: 0. 1=Enables the conversion of broadcast ExtInt and NMI interrupt requests to LINT0 and LINT1 local interrupts, respectively, before delivering to the local APIC. This conversion only takes place if the local APIC is hardware enabled. 0=ExtInt/NMI interrupts delivered unchanged.
15:12	Reserved.
11	**RespPassPW: response PassPW**. Read-write. Reset: 0. BIOS: 1. 1=The PassPW bit in all downstream responses is set, regardless of the originating request packet. This technically breaks the PCI ordering rules but it is not expected to be an issue in the downstream direction. Setting this bit improves the latency of upstream requests by allowing the downstream responses to pass posted writes. 0=The PassPW bit in downstream responses is based on the RespPassPW bit of the original request.
10:8	Reserved.
7	**CPURdRspPassPW: CPU read response PassPW**. Read-write. Reset: 0. 1=Read responses to core-generated reads are allowed to pass posted writes. 0=core responses do not pass posted writes. This bit is not expected to be set. This bit may only be set during the boot process.
6	**CPUReqPassPW: CPU request PassPW**. Read-write. Reset: 0. 1=core-generated requests are allowed to pass posted writes. 0=core requests do not pass posted writes. This bit is not expected to be set. This bit may only be set during the boot process.
5	**Cpu1En: core 1 enable**. Read-write. Reset: 0. IF (D18F3xE8[CmpCap]==0) THEN BIOS: 0. ENDIF. This bit is used to enable the CPU1 core after a reset. 1=Enable core 1 to start fetching and executing code from the boot vector.
4:0	Reserved.

D18F0x6C Link Initialization Control

Bits	Description
31:11	Reserved.
10:9	**BiosRstDet[2:1]: BIOS reset detect bits[2:1]**. Read-write. Cold reset: 0. See BiosRstDet[0].
8:7	Reserved.
6	**InitDet: CPU initialization command detect**. Read-write. Reset: 0. This bit may be used by software to distinguish between an INIT and a warm/cold reset by setting it to a 1 before an initialization event is generated. This bit is cleared by RESET_L but not by an INIT command.
5	**BiosRstDet[0]: BIOS reset detect bit[0]**. Read-write. Cold reset: 0. This bit, along with BiosRstDet[2:1], may be used to distinguish between a reset event generated by BIOS versus a reset event generated for any other reason by setting one or more of the bits to a 1 before initiating a BIOS-generated reset event.
4	**ColdRstDet: cold reset detect**. Read-write. Cold reset: 0. This bit may be used to distinguish between a cold versus a warm reset event by setting the bit to a 1 before an initialization event is generated.
3:0	Reserved.

3.8 Device 18h Function 1 Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space.

D18F1x00 Device/Vendor ID

Reset: 1701_1022h.

Bits	Description
31:16	**DeviceID: device ID**. Read-only.
15:0	**VendorID: vendor ID**. Read-only.

D18F1x04 Status/Command

Read-only. Reset: 0000_0000h.

Bits	Description
31:16	**Status**.
15:0	**Command**.

D18F1x08 Class Code/Revision ID

Reset: 0600_0000h.

Bits	Description
31:8	**ClassCode**. Read-only. Provides the host bridge class code as defined in the PCI specification.
7:0	**RevID: revision ID**. Read-only. Processor revision.

D18F1x0C Header Type

Bits	Description
31:0	**HeaderTypeReg**. Value: 0080_0000h. The header type field indicates that there are multiple functions present in this device.

D18F1x34 Capabilities Pointer

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**CapPtr: capabilities pointer**. Read-only.

D18F1x40 DRAM Base

Reset: 0000_0000h.
These registers specify the DRAM address range, base and limit:

Base Address	Limit Address
F1x40	F1x44

DRAM mapping rules:
- Transaction addresses are within the defined range if:

{0h, DramBase[35:24], 00_0000h} <= address[35:0] <= {0h, DramLimit[35:24], FF_FFFFh}.
- Accesses to addresses that map to both DRAM, as specified by D18F1x40, and MMIO, as specified by D18F1x[B8,B0,A8,A0,98,90,88,80], are routed to MMIO only.
- Programming of the DRAM address maps must be consistent with the Memory-Type Range Registers (MTRRs) and the top of memory registers, MSRC001_001A and MSRC001_001D. CPU accesses only hit within the DRAM address maps if the corresponding MTRR is of type DRAM. Accesses from the link are routed based on D18F1x40 [DRAM Base], only.
- The appropriate RE or WE bit(s) must be set.
- See 2.8.4.1.1 [DRAM and MMIO Memory Space].

Hoisting. When memory hoisting is enabled (D18F1xF0[DramHoleValid]=1), D18F1x44[DramLimit] should be set up to account for the memory hoisted above the hole. I.e., D18F1x44[DramLimit] should be set to D18F1x40[DramBase] plus the size of the amount of memory plus the hole size (4G minus D18F1xF0[DramHoleBase]). See 2.9.5 [Memory Hoisting].

Bits	Description
31:28	Reserved.
27:16	**DramBase[35:24]: DRAM base address register bits[35:24]**. Read-only.
15:2	Reserved.
1	**WE: write enable**. Read-write. 1=Writes to this address range are enabled.
0	**RE: read enable**. Read-write. 1=Reads to this address range are enabled.

D18F1x44 DRAM Limit

Reset: FFFF_0000h. See D18F1x40.

Bits	Description
31:28	**DramLimit[39:36]: DRAM limit address register bits[39:36]**. IF (D18F2x118[C6DramLock] == 1) THEN Read-only. ELSE Read-write. ENDIF. Reset: Fh. BIOS: 0h.
27:16	**DramLimit[35:24]: DRAM limit address register bits[35:24]**. IF (D18F2x118[C6DramLock] == 1) THEN Read-only. ELSE Read-write. ENDIF. Reset: FFFh.
15:0	Reserved.

D18F1x[B8,B0,A8,A0,98,90,88,80] Memory Mapped IO Base

These registers specify up to 8 MMIO address ranges. Each address range is specified by a base/limit register pair. The first set is F1x80 and F1x84, the second set is F1x88 and F1x8C, and so forth. Transaction addresses that are within the specified base/limit range are routed to the link. See 2.8.4 [Northbridge Routing].

MMIO mapping rules:
- Transaction addresses are within the defined range if:
  {0h, MMIOBase[35:16], 0000h} <= address[35:0] <= {0h, MMIOLimit[35:16], FFFFh}.
- MMIO regions must not overlap each other.
- Accesses to addresses that map to both DRAM, as specified by D18F1x40, and MMIO, as specified by D18F1x[B8,B0,A8,A0,98,90,88,80], are routed to MMIO only.
- Programming of the MMIO address maps must be consistent with the Memory-Type Range Registers (MTRRs) and the top of memory registers, MSRC001_001A and MSRC001_001D. CPU accesses only hit within the MMIO address maps if the corresponding MTRR is of type IO. Accesses from the link routed are

based on D18F1x[B8,B0,A8,A0,98,90,88,80], only.
- MMIO configuration accesses are not affected by MMIO ranges. See MSRC001_0058.
- When initializing a base/limit pair, the BIOS must write the limit register (F1x[BC,B4,AC,A4,9C,94,8C,84]) before either the RE or WE bit is set. When changing a base/limit pair that is already enabled, the BIOS should clear RE and WE before changing the address range.
- See 2.8.4.1.1 [DRAM and MMIO Memory Space].

Bits	Description
31:28	**MMIOBase[39:36]: MMIO base address register bits[39:36]**. Read-write. Reset: X. BIOS: 0h.
27:8	**MMIOBase[35:16]: MMIO base address register bits[35:16]**. Read-write. Reset: X.
7:4	Reserved.
3	**Lock**. Read-write. Reset: X. 1=D18F1x[B8,B0,A8,A0,98,90,88,80] and D18F1x[BC,B4,AC,A4,9C,94,8C,84], are read-only (including this bit). WE or RE in this register must be set in order for this to take effect.
2	Reserved.
1	**WE: write enable**. Read-write. Reset: 0. 1=Writes to this address range are enabled.
0	**RE: read enable**. Read-write. Reset: 0. 1=Reads to this address range are enabled.

D18F1x[BC,B4,AC,A4,9C,94,8C,84] Memory Mapped IO Limit

See D18F1x[B8,B0,A8,A0,98,90,88,80].

Bits	Description
31:28	**MMIOLimit[39:36]: MMIO limit address register bits[39:36]**. Read-write. Reset: X. BIOS: 0h
27:8	**MMIOLimit[35:16]: MMIO limit address register bits[35:16]**. Read-write. Reset: X.
7	**NP: non-posted**. Read-write. Reset: X. 1=CPU write requests to this MMIO range are passed through the non-posted channel. This may be used to force writes to be non-posted for MMIO regions which map to the legacy ISA/LPC bus, or in conjunction with D18F0x68 [Link Transaction Control][DsNpReqLmt] in order to allow downstream CPU requests to be counted and thereby limited to a specified number. This latter use of the NP bit may be used to avoid loop deadlock scenarios in systems that implement a region in an IO device that reflects downstream accesses back upstream. See the link specification summary of deadlock scenarios for more information. 0=CPU writes to this MMIO range use the posted channel. This bit does not affect requests that come from the link (the virtual channel of the request is specified by the link request). If two MMIO ranges target the same IO device and the NP bit is set differently in both ranges, unexpected transaction ordering effects are possible. In particular, using PCI- and IO-link-defined producer-consumer semantics, if a producer (e.g., the processor) writes data using a non-posted MMIO range followed by a flag to a posted MMIO range, then it is possible for the device to see the flag updated before the data is updated.
6:0	Reserved.

D18F1xC0 IO-Space Base

This register and D18F1xC4 specify positive decode for IO addresses to the link for transactions resulting from x86-defined IN and OUT instructions. The IO address range is specified by 1 set of base/limit registers. See 2.8.4 [Northbridge Routing].

IO mapping rules:
- IO-space transaction addresses are within the defined range if: {IOBase[24:12], 000h} <= address <= {IOLimit[24:12], FFFh} and as specified by the IE bit; or if the address is in the range specified by the VE bits.
- The appropriate RE or WE bit(s) must be set.
- See 2.8.4.1.2 [IO Space].

Bits	Description
31:25	Reserved.
24:12	**IOBase[24:12]: IO base address register bits[24:12]**. Read-write. Reset: X.
11:6	Reserved.
5	**IE: ISA enable**. Read-write. Reset: X. 1=The IO-space address window is limited to the first 256 bytes of each 1K byte block specified; this only applies to the first 64K bytes of IO space. 0=The PCI IO window is not limited in this way.
4	**VE: VGA enable**. Read-write. Reset: X. 1=Include IO-space transactions targeting the VGA-compatible address space within the IO-space window of this base/limit pair. These include IO accesses in which address bits[9:0] range from 3B0h to 3BBh or 3C0h to 3DFh (address bits[15:10] are not decoded); this only applies to the first 64K of IO space; i.e., address bits[24:16] must be low). 0=IO-space transactions targeting VGA-compatible address ranges are not added to the IO-space window. The MMIO range associated with the VGA enable bit in the PCI specification is not included in the VE bit definition; to map this range to a link, see D18F1xF4 [VGA Enable]. When D18F1xF4[VE] is set, the state of this bit is ignored.
3:2	Reserved.
1	**WE: write enable**. Read-write. Reset: 0. 1=Writes to this IO-space address range are enabled.
0	**RE: read enable**. Read-write. Reset: 0. 1=Reads to this IO-space address range are enabled.

D18F1xC4 IO-Space Limit

For detailed description see D18F1xC0 and 2.8.4.1.2 [IO Space].

Bits	Description
31:25	Reserved.
24:12	**IOLimit[24:12]: IO limit address register bits[24:12]**. Read-write. Reset: X.
11:0	Reserved.

D18F1xF0 DRAM Hole Address

Reset: 0000_0000h.

Bits	Description
31:24	**DramHoleBase[31:24]: DRAM hole base address**. IF (D18F2x118[C6DramLock] == 1) THEN Read-only. ELSE Read-write. ENDIF. This specifies the base address of the IO hole, below the 4G address level, that is used in memory hoisting. Normally, DramHoleBase >= MSRC001_001A[TOM[31:24]]. DramHoleBase must be > 0. See 2.9.5 [Memory Hoisting] for additional programming information.
23:16	Reserved.

15:7	**DramHoleOffset[31:23]: DRAM hole offset address**. IF (D18F2x118[C6DramLock] == 1) THEN Read-only. ELSE Read-write. ENDIF. When memory hosting is enabled, this value is subtracted from the physical address of certain transactions before being passed to the DCT. See 2.9.5 [Memory Hoisting] for additional programming information.
6:1	Reserved.
0	**DramHoleValid**. IF (D18F2x118[C6DramLock] == 1) THEN Read-only. ELSE Read-write. ENDIF. 1=Memory hoisting is enabled. 0=Memory hoisting is not enabled. See 2.9.5 [Memory Hoisting] for additional programming information.

D18F1xF4 VGA Enable

Reset: 0000_0000h. Read-write. All these bits are read-write unless Lock is set.

Bits	Description
31:4	Reserved.
3	**Lock**. Read-write. 1=All the bits in this register are read-only.
2	**CpuDis: CPU Disable**. Read-write. 1=The D18F1xF4[VE]-defined MMIO range is disabled for CPU accesses; i.e., CPU accesses to this range are treated as if the VE=0.
1	**NP: non-posted**. Read-write. 1=CPU write requests to the D18F1xF4[VE]-defined MMIO range are passed through the non-posted channel. 0=CPU writes may be posted.
0	**VE: VGA enable**. Read-write. 1=Transactions targeting the VGA-compatible address space are routed and controlled as specified by this register. 0=Transactions targeting the VGA-compatible address space are not affected by the state of this register. • The VGA-compatible address space is: (1) the MMIO range A_0000h through B_FFFFh; (2) IO-space accesses in which address bits[9:0] range from 3B0h to 3BBh or 3C0h to 3DFh (address bits[15:10] are not decoded; this only applies to the first 64K of IO space; i.e., address bits[24:16] must be low). • An MMIO range (F1x[BC:80]) must not overlap the VGA-compatible address space when F1xF4[VE]=1. • When this bit is set, the state of D18F1xC0[VE] is ignored.

3.9 Device 18h Function 2 Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space.

D18F2x00 Device/Vendor ID

Reset: 1702_1022h.

Bits	Description
31:16	**DeviceID: device ID**. Read-only.
15:0	**VendorID: vendor ID**. Read-only.

D18F2x04 Status/Command

Read-only. Reset: 0000_0000h.

Bits	Description
31:16	**Status**.
15:0	**Command**.

D18F2x08 Class Code/Revision ID

Reset: 0600_0000h.

Bits	Description
31:8	**ClassCode**. Read-only. Provides the host bridge class code as defined in the PCI specification.
7:0	**RevID: revision ID**. Read-only.

D18F2x0C Header Type

Bits	Description
31:0	**HeaderTypeReg**. Value: 0080_0000h.The header type field indicates that there are multiple functions present in this device.

D18F2x34 Capabilities Pointer

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**CapPtr: capabilities pointer**. Read-only.

D18F2x[4C:40] DRAM CS Base Address

Reset: 0000_0000h. See 2.9.1 [DCT Configuration Registers] for general programming information about DCT configuration registers.

These registers along with D18F2x[64:60] [DRAM CS Mask], translate DRAM request addresses (to a DRAM controller) into DRAM chip selects. Supported DIMM sizes are specified in D18F2x80 [DRAM Bank Address Mapping]. See 2.9 [DRAM Controller (DCT)].

For each chip select, there is a DRAM CS Base Address register. For every two chip selects there is a DRAM CS Mask register. These are associated with DIMM numbers, CKE, and ODT signals as follows:

Table 52: DIMM, Chip Select, CKE, ODT, and Register Mapping

Base Address Registers	Mask Register	DIMM Number	Chip Select	CKE	ODT
D18F2x40	D18F2x60	0	MA0_CS_L[0]	MA_CKE[0]	MA0_ODT[0]
D18F2x44			MA0_CS_L[1]	MA_CKE[1]	MA0_ODT[1]
D18F2x48	D18F2x64	1	MA1_CS_L[0]	MA_CKE[0]	MA1_ODT[0]
D18F2x4C			MA1_CS_L[1]	MA_CKE[1]	MA1_ODT[1]

The DRAM controller operates on the normalized physical address of the DRAM request. The normalized physical address includes all of the address bits that are supported by a DRAM controller. See 2.8.1 [Northbridge (NB) Architecture].

Each base address register specifies the starting normalized address of the block of memory associated with the chip select. Each mask register specifies the additional address bits that are consumed by the block of memory associated with the chip selects. If both chip selects of a DIMM are used, they must be the same size; in this case, a single mask register covers the address space consumed by both chip selects.

Lower-order address bits are provided in the base address and mask registers, as well. These allow memory to be interleaved between chip selects, such that contiguous physical addresses map to the same DRAM page of multiple chip selects. See 2.9.4 [Chip Select Interleaving].

System BIOS is required to assign the largest DIMM chip select range to the lowest normalized address of the DRAM controller. As addresses increase, the chip-select size is required to remain constant or decrease. This is necessary to keep DIMM chip select banks on aligned address boundaries, regardless as to the amount of address space covered by each chip select.

For each normalized address for requests that enters a DRAM controller, a ChipSelect[i] is asserted if:

```
CSEnable[i] &
   ( {(InputAddr[35:27]  & ~AddrMask[i][35:27]),
      (InputAddr[21:13]  & ~AddrMask[i][21:13])} ==
     {(BaseAddr[35:27]    & ~AddrMask[i][35:27]),
      (BaseAddr[21:13]    & ~AddrMask[i][21:13])} );
```

Bits	Description
31:29	Reserved.
28	Reserved.
27:19	**BaseAddr[35:27]: normalized physical base address bits [35:27]**. IF (D18F2x118[C6DramLock] == 1) THEN Read-only. ELSE Read-write. ENDIF.
18:14	Reserved.
13:5	**BaseAddr[21:13]: normalized physical base address bits [21:13]**. IF (D18F2x118[C6DramLock] == 1) THEN Read-only. ELSE Read-write. ENDIF.
4	Reserved.

3	**OnDimmMirror: on-DIMM mirroring (ODM) enabled**. IF (D18F2x118[C6DramLock] == 1) THEN Read-only. ELSE Read-write. ENDIF. 1=Address and bank bits are swapped by hardware for MRS commands sent to this chip select. This mode accounts for routing on the DIMM. This bit is expected to be set for the odd numbered rank of unbuffered DDR3 DIMMs if SPD byte 63 indicates that address mapping is mirrored. Hardware bit swapping does not occur for commands sent via D18F2x7C[SendMrsCmd] when D18F2x7C[EnDramInit] = 0. See 2.9.3.6.1.1 [DDR3 MR Initialization]. The following bits are swapped when enabled: • M[B, A]_BANK[0] and M[B, A]_BANK[1]. • M[B, A]_ADD[3] and M[B, A]_ADD[4]. • M[B, A]_ADD[5] and M[B, A]_ADD[6]. • M[B, A]_ADD[7] and M[B, A]_ADD[8]. This bit must be programmed properly before initializing the DRAM devices.
2	**TestFail: memory test failed**. IF (D18F2x118[C6DramLock] == 1) THEN Read-only. ELSE Read-write. ENDIF. Set by BIOS to indicate that a rank is present but has failed memory training or a memory consistency test, indicating that the memory is bad. BIOS should treat CSEnable=1 and TestFail=1 as mutually exclusive.
1	Reserved.
0	**CSEnable: chip select enable**. IF (D18F2x118[C6DramLock] == 1) THEN Read-only. ELSE Read-write. ENDIF.

D18F2x[64:60] DRAM CS Mask

Reset: 0000_0000h. See 2.9.1 [DCT Configuration Registers] for general programming information about DCT configuration registers. See D18F2x[4C:40] for information about this register.

Bits	Description
31:29	Reserved.
28	Reserved.
27:19	**AddrMask[35:27]: normalized physical address mask bits [35:27]**. IF (D18F2x118[C6DramLock] == 1) THEN Read-only. ELSE Read-write. ENDIF.
18:14	Reserved.
13:5	**AddrMask[21:13]: normalized physical address mask bits [21:13]**. IF (D18F2x118[C6DramLock] == 1) THEN Read-only. ELSE Read-write. ENDIF.
4:0	Reserved.

D18F2x78 DRAM Control

See 2.9.1 [DCT Configuration Registers] for general programming information about DCT configuration registers.

Bits	Description
31:22	**MaxRdLatency: maximum read latency**. Read-write. Reset: 12h. BIOS: See 2.9.3.7.4. This field should be programmed by the system BIOS to specify the maximum round-trip latency in the system from the processor to the DRAM devices and back. The DRAM controller uses this field to determine when incoming DRAM read data can be safely transferred to the DRAM controller clock (NCLK) domain. The time includes the asynchronous and synchronous latencies. Bits — Definition 004h-000h — Reserved 050h-005h — <MaxRdLatency> NCLKs 3FFh-051h — Reserved
21	**DisCutThroughMode: disable cut through mode**. Read-write. Reset: 1. BIOS: 0. 1=DRAM controller waits for all 512 bits of DRAM read data before transferring data to DRAM controller clock (NCLK) domain. 0=DRAM controller waits for first 256 bits of DRAM read data before transferring. See 2.9.3.7.4 [Calculating MaxRdLatency].
20	**ForceCasToSlot0: force CAS to slot 0**. Read-write. Reset: 0. BIOS: See 2.9.3.7.4.1. 1=DRAM controller will issue all CAS commands in slot 0 of an unskipped DRAM controller clock (NCLK). 0=DRAM controller can issue CAS commands in either slot 0 or slot 1. See 2.9.3.7.4.1 [MaxRdLatency Training].
19:18	Reserved.
17	**AddrCmdTriEn: address command tri-state enable**. Read-write. Reset: 0. BIOS: See 2.9.3.5. 1=Tri-state the address, command, and bank pins when a Deselect command is issued.
16:14	Reserved.
13:12	**Trdrd[3:2]: read to read timing**. Read-write. Reset: 0. This field along with D18F2x8C[Trdrd[1:0]] combine to specify a 4-bit value, Trdrd[3:0]. See: D18F2x8C[Trdrd[1:0]].
11:10	**Twrwr[3:2]: write to write timing**. Read-write. Reset: 0. This field along with D18F2x8C[Twrwr[1:0]] combine to specify a 4-bit value, Twrwr[3:0]. See: D18F2x8C[Twrwr[1:0]].
9:8	**Twrrd[3:2]: write to read DIMM termination turnaround**. Read-write. Reset: 0. This field along with D18F2x8C[Twrrd[1:0]] combine to specify a 4-bit value, Twrrd[3:0]. See: D18F2x8C[Twrrd[1:0]].
7	Reserved.
6	**RxPtrInitReq: receive FIFO pointer initialization request**. Read; write-1-only; cleared-when-done. Reset: 0. 1=The DCT performs receive FIFO pointer initialization. This bit is cleared by hardware after the initialization completes.

5:4	Reserved.

3:0	**RdPtrInit: read pointer initial value**. Read-write. Reset: 2h. BIOS: See 2.9.3.2.2. There is a synchronization FIFO between the NB clock domain and the memory clock domain. This field specifies where the read pointer is placed at the time of FIFO initialization. This field along with D18F2x94[MemClkFreq] gives an offset from the minimum read to write pointer separation as follows:
	IF (D18F2x94[MemClkFreq] >= 667 MHz) THEN
	Bits — Definition
	0000b — Minimum separation + 1.0 MEMCLKs
	0001b — Minimum separation + 0.5 MEMCLK
	0010b — Minimum separation
	1110b-0011b — Reserved
	1111b — IF (D18F2xA8[DbeGskMemClkAlignMode]==01b) THEN Reserved ELSE Minimum separation + 1.5 MEMCLKs ENDIF.
	ELSE
	Bits — Definition
	0000b — IF (D18F2xA8[DbeGskMemClkAlignMode]==01b) THEN Reserved ELSE Minimum separation + 1.5 MEMCLKs ENDIF.
	0001b — Minimum separation + 1.0 MEMCLKs
	0010b — Minimum separation + 0.5 MEMCLK
	0011b — Minimum separation
	1111b-0100b — Reserved
	ENDIF.

D18F2x7C DRAM Initialization

Reset: 0000_0000h.

BIOS can directly control the DRAM initialization sequence using this register. To do so, BIOS sets EnDramInit to start DRAM initialization. BIOS should then complete the initialization sequence specified in the appropriate JEDEC specification. After completing the sequence, BIOS clears EnDramInit to complete DRAM initialization.

Setting more than one of the command bits in this register (SendZQCmd, SendMrsCmd, SendAutoRefresh, and SendPchgAll) at a time results in undefined behavior. See 2.9.3 [DCT/DRAM Initialization and Resume].

Bits	Description
31	**EnDramInit: enable DRAM initialization**. Read-write. BIOS: See 2.9.3.2 and 2.9.3.6.1. 1=Place the DRAM controller in BIOS-controlled DRAM initialization mode. The DCT asserts memory reset and de-asserts CKE when this bit is set.
30	Reserved.
29	**SendZQCmd: send ZQ command**. Read; write-1-only; cleared-when-done. 1=The DCT sends the ZQ calibration command to all enabled chip selects. This bit is cleared by the hardware after the command completes.
28	**AssertCke: assert CKE**. Read-write. Setting this bit causes the DCT to assert the CKE pins. This bit cannot be used to de-assert the CKE pins.

27	**DeassertMemRstX: de-assert memory reset**. Read-write. Setting this bit causes the DCT to de-assert the memory reset. This bit cannot be used to assert the memory reset pin.
26	**SendMrsCmd: send MRS/EMRS command**. Read; Write-1-only; cleared-when-done. 1=The DCT sends the MRS or EMRS commands defined by the MrsAddress and MrsBank fields of this register. Software is responsible for DRAM timing parameter enforcement.
25	**SendAutoRefresh: send auto refresh command**. Read; Write-1-only; cleared-when-done. 1=The DCT sends an auto refresh command. Only valid when EnDramInit=1.
24	**SendPchgAll: send precharge all command**. Read; Write-1-only; cleared-when-done. 1=The DCT sends a precharge-all command. Only valid when EnDramInit=1.
23	Reserved.
22:20	**MrsChipSel: MRS/EMRS command chip select**. Read-write. This field specifies which DRAM chip select is used for MRS/EMRS commands. This field is valid only when EnDramInit = 0 and when SendMrsCmd=1; otherwise, MRS/EMRS commands are sent to all chip selects. Bits — Definition 000b — MRS/EMRS command is sent to CS0 001b — MRS/EMRS command is sent to CS1 010b — MRS/EMRS command is sent to CS2 011b — MRS/EMRS command is sent to CS3 111b-100b — Reserved.
19	Reserved.
18:16	**MrsBank: bank address for MRS/EMRS commands**. Read-write. This field specifies the data driven on the DRAM bank pins for MRS and EMRS commands.
15:0	**MrsAddress: address for MRS/EMRS commands**. Read-write. This field specifies the data driven on the DRAM address pins 15-0 for MRS and EMRS commands.

D18F2x80 DRAM Bank Address Mapping

Reset: 0000_0000h.

See 2.9.1 [DCT Configuration Registers] for general programming information about DCT configuration registers.

These fields specify DIMM configuration information. These fields are required to be programmed per the following table, based on the DRAM device size and width information of the DIMM. Table 53 shows the bit numbers for each position.

Bits	Description
31:8	Reserved.
7:4	**Dimm1AddrMap: DIMM 1 address map**. IF (D18F2x118[C6DramLock] == 1) THEN Read-only. ELSE Read-write. ENDIF.
3:0	**Dimm0AddrMap: DIMM 0 address map**. IF (D18F2x118[C6DramLock] == 1) THEN Read-only. ELSE Read-write. ENDIF.

Table 53: DDR3 DRAM address mapping

Bits	CS Size	Device size, width	Bank			Address																
			2	1	0		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0000b		Reserved				Row																
						Col																
0001b	256MB	512Mb, x16	15	14	13	Row	x	x	x	x	17	16	27	26	25	24	23	22	21	20	19	18
						Col	x	x	x	x	x	AP	12	11	10	9	8	7	6	5	4	3
0010b	512MB	512Mb, x8	15	14	13	Row	x	x	x	17	16	28	27	26	25	24	23	22	21	20	19	18
		1Gb, x16				Col	x	x	x	x	x	AP	12	11	10	9	8	7	6	5	4	3
0011b		Reserved				Row																
						Col																
0100b		Reserved				Row																
						Col																
0101b	1GB	1Gb, x8	15	14	13	Row	x	x	17	16	29	28	27	26	25	24	23	22	21	20	19	18
		2Gb, x16				Col	x	x	x	x	x	AP	12	11	10	9	8	7	6	5	4	3
0110b		Reserved				Row	x															
						Col	x															
0111b	2GB	2Gb, x8	15	14	13	Row	x	17	16	30	29	28	27	26	25	24	23	22	21	20	19	18
		4Gb, x16				Col	x	x	x	x	x	AP	12	11	10	9	8	7	6	5	4	3
1000b		Reserved				Row																
						Col																
1001b		Reserved				Row																
						Col																
1010b	4GB	4Gb, x8	15	14	13	Row	17	16	31	30	29	28	27	26	25	24	23	22	21	20	19	18
		8Gb, x16				Col	x	x	x	x	x	AP	12	11	10	9	8	7	6	5	4	3
1011b	8GB	8Gb, x8	16	15	14	Row	17	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18
						Col	x	x	x	x	13	AP	12	11	10	9	8	7	6	5	4	3

D18F2x84 DRAM MRS

Reset: 0000_0001h.

All the fields of this register are programmed into the DRAM device mode registers, MR[3, 2, 1, 0], for each DRAM device during the DRAM initialization process.

Bits	Description
31:24	Reserved.
23	**PchgPDModeSel: precharge power down mode select**. Read-write. BIOS: 1. 0=DDR3-defined slow exit mode; the DCT issues the first valid read, read with auto-precharge, or synchronous ODT command a minimum of Txpdll after precharge power down exit. 1=DDR3-defined fast exit mode; the DCT issues the first valid command a minimum of Txp after precharge power down exit.
22:20	**Tcwl: CAS write latency**. Read-write. This specifies the number of clock cycles from internal write command to first write data in. Bits Definition Bits Definition 100b-000b <Tcwl+5> clocks 111b-101b Reserved
19:7	Reserved.

6:4	**Twr: write recovery**. Read-write. BIOS: See 2.9.3.3. This specifies the minimum time from the last data write until the chip select bank precharge; this is the WR field in the DDR3 specification. Bits Definition Bits Definition 000b 16 MEMCLK cycles 100b 8 MEMCLK cycles 001b 5 MEMCLK cycles 101b 10 MEMCLK cycles 010b 6 MEMCLK cycles 110b 12 MEMCLK cycles 011b 7 MEMCLK cycles 111b 14 MEMCLK cycles
3:2	Reserved.
1:0	**BurstCtrl: burst length control**. Read-write. Specifies the number of sequential beats of DQ related to one read or write command. Bits Definition 00b Reserved. 01b 4 or 8-beat burst length on the fly; one 32-byte access or one 64-byte access 1xb Reserved

D18F2x88 DRAM Timing Low

Reset: FF00_0001h.

See 2.9.1 [DCT Configuration Registers] for general programming information about DCT configuration registers.

Bits	Description
31:24	**MemClkDis: MEMCLK disable**. Read-write. BIOS: See 2.9.3.8. 1=Disable the MEMCLK. The bits MemClkDis[7:0] are mapped to package pin names as follows: Bit Package pin name [0] MA_CLK_H/L[0]. [1] MA_CLK_H/L[1]. [2] MA_CLK_H/L[2]. [3] MA_CLK_H/L[3]. [7:4] Reserved
23:4	Reserved.
3:0	**Tcl: CAS latency**. Read-write. BIOS: See 2.9.3.3. This specifies the time from the CAS assertion for a read cycle until data return (from the perspective of the DRAM devices). Bits Definition 0h Reserved Ah-1h <Tcl+4> clocks Fh-Bh Reserved

D18F2x8C DRAM Timing High

Reset: 0000_000Ah.

See 2.9.1 [DCT Configuration Registers] for general programming information about DCT configuration registers.

Bits	Description
31:26	Reserved.

25:23	**Trfc1: auto-refresh row cycle time for DIMM1**. Read-write. See: Trfc0.
22:20	**Trfc0: auto-refresh row cycle time for DIMM0**. Read-write. BIOS: See 2.9.3.3. This specifies the minimum time from an auto-refresh command to the next valid command, except NOP or DES. DIMM numbers are specified by D18F2x[4C:40] [DRAM CS Base Address]. The recommended programming of this register varies based on DRAM density and speed. Bits Definition Bits Definition 000b Reserved 100b 300 ns (all speeds, 4 Gbit) 001b 90 ns (all speeds, 512 Mbit) 101b 350 ns (all speeds, 8 Gbit) 010b 110 ns (all speeds, 1 Gbit) 11xb Reserved 011b 160 ns (all speeds, 2 Gbit)
19	Reserved.
18	**DisAutoRefresh: disable automatic refresh**. Read-write. BIOS: See 2.9.3.5. 1=Automatic refresh is disabled. BIOS must set this bit prior to DRAM initialization and it must remain set until DRAM training has completed. Subsequent register accesses may only set this bit during S3 exit or if the DRAM has been placed into self-refresh.
17:16	**Tref: refresh rate**. Read-write. BIOS: See 2.9.3.3. This specifies the average time between refresh requests to all DRAM devices. Bits Definition 00b Undefined behavior. 01b Reserved 10b Every 7.8 us. 11b Every 3.9 us.
15:14	**Trdrd[1:0]: read-to-read timing**. Read-write. BIOS: See 2.9.3.4.1 [Trdrd and TrdrdSD (Read-to-Read Timing)]. Trdrd specifies the minimum number of cycles from the last clock of *virtual CAS* of a first read-burst operation to the clock in which CAS is asserted for a following read-burst operation to a different DIMM. This field along with D18F2x78[Trdrd[3:2]] combine to specify a 4-bit value, Trdrd[3:0]. Bits Definition 8h-0h <Trdrd+2> clocks Fh-9h Reserved
13:12	**Twrwr[1:0]: write-to-write timing**. Read-write. BIOS: See 2.9.3.4.2 [Twrwr and TwrwrSD (Write-to-Write Timing)]. Twrwr specifies the minimum number of cycles from the last clock of *virtual CAS* of the first write-burst operation to the clock in which CAS is asserted for a following write-burst operation to a different DIMM. This field along with D18F2x78[Twrwr[3:2]] combine to specify a 4-bit value, Twrwr[3:0]. Bits Definition 9h-0h <Twrwr+1> clocks Fh-Ah Reserved
11:10	**Twrrd[1:0]: write-to-read DIMM termination turnaround**. Read-write. BIOS: See 2.9.3.4.3 [Twrrd and TwrrdSD (Write-to-Read DIMM Termination Turn-around)]. This specifies the minimum number of cycles from the last clock of *virtual CAS* of the first write operation to the clock in which CAS is asserted for a following read operation to a different DIMM. This field along with D18F2x78[Twrrd[3:2]] combine to specify a 4-bit value, Twrrd[3:0]. Bits Definition Ah-0h <Twrrd+1> clocks Fh-Bh Reserved
9:8	Reserved.

7:4	**TrwtTO: read-to-write turnaround for data, DQS contention**. Read-write. BIOS: See 2.9.3.4.4 [TrwtTO (Read-to-Write Turnaround for Data, DQS Contention)]. This specifies the minimum number of cycles from the last clock of *virtual CAS* of a first read operation to the clock in which CAS is asserted for a following write operation. Time may need to be inserted to ensure there is no bus contention on bidirectional pins. Bits: Definition 0h: Reserved Fh-1h: <TrwtTO+2> clocks (<TrwtTO> idle cycles on data bus) See 2.9.3.4.4 [TrwtTO (Read-to-Write Turnaround for Data, DQS Contention)].
3:0	**TrwtWB: read-to-write turnaround for opportunistic write bursting**. Read-write. BIOS: 4h. Specifies the minimum number of NCLK cycles from the last read operation seen by the DCT scheduler to the following write operation. The purpose of this field is to hold off write operations until several cycles have elapsed without a read cycle; this may result in a performance benefit. If opportunistic write bursting is disabled then DCT write bursting should be disabled by setting D18F2x11C[DctWrLimit] to 1Fh. Bits: Definition 0h: Disabled Fh-1h: <TrwtWB> NCLKs

D18F2x90 DRAM Configuration Low

See 2.9.1 [DCT Configuration Registers] for general programming information about DCT configuration registers.

Bits	Description
31:28	Reserved.
27	**DisDllShutdownSR: disable DLL shutdown in self-refresh mode**. Read-write. Reset: 0. BIOS: 0. See 2.5.5.1 [DRAM Self-Refresh]. 1=DDR phy DLLs remain active during DRAM self refresh. 0=DDR phy DLLs are shutdown during self-refresh.
26	**DbeSkidBufDis: disable skid buffer**. Read-write. Reset: 0. BIOS: See 2.9.3.5. 1=Disable the performance enhancing skid buffer and arbitrate normally. The skid buffer allows the arbiter to pick a lower relative priority page miss ahead of a page hit, so that Trcd penalty for subsequent CAS is hidden behind the ready/requested CAS.
25	**EnDispAutoPrecharge: enable auto-precharge for display traffic**. Read-write. Reset: 0. BIOS: 1. 1=Auto-precharge commands are generated for the last read or write of a display burst if no other commands are pending to the DRAM page.
24	Reserved.
23	**ForceAutoPchg: force auto-precharging**. Read-write. Reset: 0. BIOS: 0. 1=Force auto-precharge cycles with every read or write command. This may be preferred in situations where power savings is favored over performance.
22:21	**IdleCycInit: idle cycle counter initial value**. Read-write. Reset: 0. BIOS: 11b. This specifies the initial number of MEMCLK cycles during which an open page of DRAM is not accessed before it may be closed by the dynamic page close logic. This field is ignored if D18F2x90[DynPageCloseEn] = 0. Bits: Definition: Bits: Definition 00b: 16 clocks: 10b: 64 clocks 01b: 32 clocks: 11b: 96 clocks

20	**DynPageCloseEn: dynamic page close enable**. Read-write. Reset: 0. BIOS: 0. 1=The DRAM controller dynamically determines when to close open pages based on the history of that particular page and D18F2x90[IdleCycInit]. 0=Any open pages not auto-precharged by the DRAM controller are automatically closed after a time controlled by D18F6x74.
19:18	Reserved.
17	**EnterSelfRef: enter self refresh command**. Read; write-1-only; cleared-by-hardware. Reset: 0. 1=Command the DRAMs to enter into self refresh mode. 0=The enter-self-refresh command has completed executing. See DisDllShutdownSR and 2.9.3.7 [DRAM Training]. NB P-state transitions must be disabled prior to setting this bit. See D18F6x90[NbPsCtrlDis] and See 2.5.4.1.3 [Software Controlled NB P-states].
16:2	Reserved.
1	**ExitSelfRef: exit self refresh command**. Read; write-1-only; cleared-by-hardware. Reset: 0. 1=Command the DRAM controller to bring the DRAMs out of self refresh mode. 0=The exit-self-refresh command has completed. This command should be executed by BIOS when returning from the suspend to RAM state after the DRAM controller configuration registers are properly initialized (see 2.5.6.1.1 [ACPI Suspend to RAM State (S3)]), or when self refresh is used during DRAM training (see DisDllShutdownSR and 2.9.3.7 [DRAM Training]). This bit should not be set if the DCT is disabled.
0	Reserved.

D18F2x94 DRAM Configuration High

Reset: 0F00_0000h.

See 2.9.1 [DCT Configuration Registers] for general programming information about DCT configuration registers.

Bits	Description
31:28	**FourActWindow: four bank activate window**. Read-write. BIOS: See 2.9.3.3.1. Specifies the rolling tFAW window during which no more than 4 banks in an 8-bank device are activated. Bits — Window size 0h — No tFAW window restriction. Dh-1h — <2*FourActWindow+14> MEMCLK cycles Fh-Eh — Reserved
27:24	**DcqBypassMax: DRAM controller queue bypass maximum**. Read-write. BIOS: Eh. The DRAM controller arbiter normally allows transactions to pass other transactions in order to optimize DRAM bandwidth. This field specifies the maximum number of times that the oldest memory-access request in the DRAM controller queue may be bypassed before the arbiter decision is overridden and the oldest memory-access request is serviced instead. Bits — Definition 0h — Reserved. Eh-1h — The oldest request may be bypassed no more than <4*DcqBypassMax+4> times. Fh — The bypass maximum control is disabled.
23	**ProcOdtDis: processor on-die termination disable**. Read-write. 1=The processor-side on-die termination is disabled. 0=Processor-side on-die termination enabled. See D18F2x9C_x0000_0000[ProcOdt] for ODT definitions.

22	**BankSwizzleMode: bank swizzle mode**. Read-write. BIOS: See 2.9.3.5. 1=Remaps the DRAM device bank address bits as a function of normalized physical address bits. Each of the bank address bits, as specified in Table 53 of D18F2x80, are remapped as follows: Define X as a bank address bit (e.g., X=15 if the bank bit is specified to be address bit 15). Define S(n) as the state of address bit n (0 or 1) and B as the remapped bank address bit. Then, B= S(X) ^ S(X + 3) ^ S(X + 6); for an 8-bank DRAM. For example, encoding 02h of Table 53 would be remapped from bank[2:0]={A15, A14, A13} to the following for a 64-bit DCT: Bank[2:0] = {A15 ^ A18 ^ A21, A14 ^ A17 ^ A20, A13 ^ A16 ^ A19}.
21	Reserved.
20	**SlowAccessMode: slow access mode (also known as 2T mode)**. Read-write. BIOS: Table 21 through Table 22. 1=One additional MEMCLK of setup time is provided on all DRAM address and control signals (not including CS, CKE, and ODT); i.e., these signals are driven for two MEMCLK cycles rather than one. 0=DRAM address and control signals are driven for one MEMCLK cycle. 2T mode may be needed in order to meet electrical requirements of certain DIMM speed and loading configurations.
19:17	Reserved.
16	**PowerDownMode: power down mode**. Read-write. BIOS: 1. This specifies how a DIMM enters power down mode, when enabled by D18F2x94[PowerDownEn]. A DIMM enters power down mode when the DCT de-asserts the CKE pin to that DIMM. The command and address signals tri-state one MEMCLK after CKE de-asserts. Bit Description 0b Channel CKE control mode. The DRAM channel is placed in power down mode when all chip selects associated with the channel are idle. Both CKE pins for the channel operate in lock step, in terms of placing the channel DIMMs in power down mode. 1b Chip select CKE control mode. A chip select or pair of chip selects is placed in power down mode when no transactions are pending for the chip select(s). This mode is expected to be used in mobile systems: - CKE0 is associated with CS0 in 2-rank systems. - CKE1 is associated with CS1 in 2-rank systems. • See D18F2x[4C:40].
15	**PowerDownEn: power down mode enable**. Read-write. BIOS: See 2.9.3.5. 1=Power down mode is enabled. When in power down mode, if all pages of the DRAMs associated with a CKE pin are closed, then these parts are placed in power down mode. Only precharge power down mode is supported, not active power down mode.
14:12	Reserved.
11:10	**ZqcsInterval: ZQ calibration short interval**. Read-write. BIOS: See 2.9.3.5. This field specifies the programmable interval for the controller to send out the DRAM ZQ calibration short command. Bits Definition 00b ZQ calibration short command is disabled 01b 64 ms 10b 128 ms 11b 256 ms
9:8	Reserved.

7	**MemClkFreqVal: memory clock frequency valid**. Read-write. System BIOS should set this bit when setting up D18F2x94[MemClkFreq] to the proper value. This indicates to the DRAM controller that it may start driving MEMCLK at the proper frequency. This bit should not be set if the DCT is disabled. See 2.9.3.6.
6:5	Reserved.
4:0	**MemClkFreq: memory clock frequency**. Read-write. Reset: 00000b. This field specifies the frequency and rate of the DRAM interface (MEMCLK). The rate defined below is twice the frequency. See D18F3xE8[DdrMaxRate]. Bits — Definition 00101b-00000b — Reserved 00110b — 400 MHz, 800 MT/s 01001b-00111b — Reserved 01010b — 533 MHz, 1066 MT/s 11111b-01011b — Reserved

D18F2x98 DRAM Controller Additional Data Offset

Reset: 8000_0000h.

The DCT includes an array of registers that are used primarily to control DRAM-interface electrical parameters. Access to these registers is accomplished as follows:

- Reads:
  - Write the register number to D18F2x98[DctOffset] with D18F2x98[DctAccessWrite]=0.
  - Poll until D18F2x98[DctAccessDone]=1.
  - Read the register contents from D18F2x9C.

- Writes:
  - Write all 32 bits to the register data to D18F2x9C (individual byte writes are not supported).
  - Write the register number to D18F2x98[DctOffset] with D18F2x98[DctAccessWrite]=1.
  - Poll until D18F2x98[DctAccessDone]=1. This ensures that the contents of the write have been delivered to the phy.

See 2.9.1 [DCT Configuration Registers] for general programming information about DCT configuration registers.

Bits	Description
31	**DctAccessDone: DRAM controller access done**. Read-only. 1=The access to one of the D18F2x9C_x registers is complete. 0=The access is still in progress.
30	**DctAccessWrite: DRAM controller read/write select**. Read-write. 0=Read one of the D18F2x9C_x registers. 1=Write one of the D18F2x9C_x registers.
29:0	**DctOffset: DRAM controller offset**. Read-write.

D18F2x9C DRAM Controller Additional Data Port

See D18F2x98.

Bits	Description
31:0	**DctDataPort**. Read-write. See D18F2x98 for details about this port.

D18F2x9C_x0000_0000 DRAM Output Driver Compensation Control

Cold reset: 3033_3333h. BIOS: Table 21 through Table 22. See 2.9.3.4.6 [DRAM Address Timing and Output Driver Compensation Control].

Bits	Description
31	Reserved.
30:28	**ProcOdt: processor on-die termination**. Read-write. This field specifies the resistance of the on-die termination resistors for the DQ and DQS pins. This field is valid only when D18F2x94[ProcOdtDis]=0. If the DQ and DQS pins require different on-die termination values, then D18F2x9C_x0D0F_0[F,7:0]0[8,0][ProcOdtDqOvrd] must be programmed after any write to this register. Bits — Definition 000b — 240 ohms +/- 20% 001b — 120 ohms +/- 20% 010b — 80 ohms +/- 20% 011b — 60 ohms +/- 20% 111b-100b — Reserved.
27:23	Reserved.
22:20	**DqsDrvStren: DQS drive strength**. Read-write. This field specifies the drive strength of the DQS pins. Bits — Definition 000b — 0.75x 001b — 1.0x 010b — 1.25x 011b — 1.5x 111b-100b — Reserved
19	Reserved.
18:16	**DataDrvStren: data drive strength**. Read-write. This field specifies the drive strength of the DRAM data pins. Bits — Definition 000b — 0.75x 001b — 1.0x 010b — 1.25x 011b — 1.5x 111b-100b — Reserved
15	Reserved.
14:12	**ClkDrvStren: MEMCLK drive strength**. Read-write. This field specifies the drive strength of the MEMCLK pins. Bits — Definition 000b — 1.0x 001b — 1.25x 010b — 1.5x 011b — 2.0x 111b-100b — Reserved
11	Reserved.

10:8	**AddrCmdDrvStren: address/command drive strength**. Read-write. This field specifies the drive strength of the address, RAS, CAS, WE, bank and parity pins. Bits Definition 000b 1.0x 001b 1.25x 010b 1.5x 011b 2.0x 111b-100b Reserved
7	Reserved.
6:4	**CsOdtDrvStren: CS/ODT drive strength**. Read-write. This field specifies the drive strength of the CS and ODT pins. Bits Definition 000b 1.0x 001b 1.25x 010b 1.5x 011b 2.0x 111b-100b Reserved
3	Reserved.
2:0	**CkeDrvStren: CKE drive strength**. Read-write. CkeDrvStren has no effect; the CKE drive strength is always 2.0x.

D18F2x9C_x0000_0[1:0]0[2:1] DRAM Write Data Timing

BIOS: See 2.9.3.7.3 [DQS Position Training]. These registers control the timing of write DQ with respect to MEMCLK and allow transmit DQS to be centered in the data eye. The delay starts 1 UI before the rise edge of MEMCLK corresponding to the CAS-write-latency. The total delay for each byte is the sum of WrDatGross-Dly and WrDatFineDly.

Table 54: Index addresses for D18F2x9C_x0000_0[1:0]0[2:1]

D18F2x98[31:16]	D18F2x98[15:0]			
	0102h	0101h	0002h	0001h
0000h	DIMM 1 Bytes 7-4	DIMM 1 Bytes 3-0	DIMM 0 Bytes 7-4	DIMM 0 Bytes 3-0

Table 55: Byte lane mapping for D18F2x9C_x0000_0[1:0]0[2:1]

Register	Bits			
	31:24	23:16	15:8	7:0
D18F2x9C_x0000_0[1:0]01	Byte 3	Byte 2	Byte 1	Byte 0
D18F2x9C_x0000_0[1:0]02	Byte 7	Byte 6	Byte 5	Byte 4

Bits	Description
31:29	**WrDatGrossDly: write data gross delay**. Read-write. Reset: 0. See: D18F2x9C_x0000_0[1:0]0[2:1][7:5].
28:24	**WrDatFineDly: write data fine delay**. Read-write. Cold reset: 0. See: D18F2x9C_x0000_0[1:0]0[2:1][4:0].

23:21	**WrDatGrossDly: write data gross delay**. Read-write. Reset: 0. See: D18F2x9C_x0000_0[1:0]0[2:1][7:5].
20:16	**WrDatFineDly: write data fine delay**. Read-write. Cold reset: 0. See: D18F2x9C_x0000_0[1:0]0[2:1][4:0].
15:13	**WrDatGrossDly: write data gross delay**. Read-write. Reset: 0. See: D18F2x9C_x0000_0[1:0]0[2:1][7:5].
12:8	**WrDatFineDly: write data fine delay**. Read-write. Cold reset: 0. See: D18F2x9C_x0000_0[1:0]0[2:1][4:0].
7:5	**WrDatGrossDly: write data gross delay**. Read-write. Reset: 0. Bits Definition 100b-000b <WrDatGrossDly*0.5> MEMCLK delay 111b-101b Reserved WrDatGrossDly must be programmed to ensure that WrDatGrossDly - D18F2x9C_x0000_00[44:30][WrDqsGrossDly] <= 0.5 MEMCLKs.
4:0	**WrDatFineDly: write data fine delay**. Read-write. Cold reset: 0. Bits Definition 1Fh-0h <WrDatFineDly>/64 MEMCLK delay

D18F2x9C_x0000_0004 DRAM Address/Command Timing Control

BIOS: Table 21 through Table 22. See 2.9.3.4.6 [DRAM Address Timing and Output Driver Compensation Control].

This register controls the timing of the address, command, chip select, ODT and clock enable pins with respect to MEMCLK. See the figure below. This register is used to adjust both the setup and hold time at the DIMM.

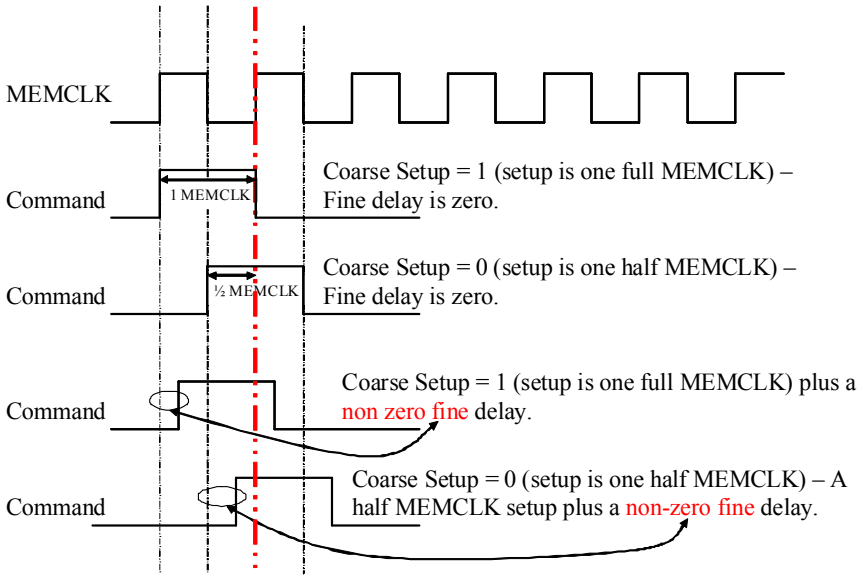


Figure 11: Address/command timing at the processor pins

2T timing is controlled by D18F2x94[SlowAccessMode].

Bits	Description
31:22	Reserved.

21	**AddrCmdSetup: address/command setup time**. Read-write. Reset: 0. This bit selects the default setup time for the address and command pins versus MEMCLK. 0=1/2 MEMCLK (1 1/2 MEMCLK for 2T timing). 1=1 MEMCLK (2 MEMCLKs for 2T timing).
20:16	**AddrCmdFineDelay: address/command fine delay**. Read-write. Cold reset: 0. This field specifies the time that the address and command pins are delayed from the default setup time. See: CkeFineDelay.
15:14	Reserved.
13	**CsOdtSetup: CS/ODT setup time**. Read-write. Reset: 0. This bit selects the default setup time for the CS and ODT pins versus MEMCLK. 0=1/2 MEMCLK. 1=1 MEMCLK
12:8	**CsOdtFineDelay: CS/ODT fine delay**. Read-write. Cold reset: 0. This field specifies the time that the CS and ODT pins are delayed from the default setup time. See: CkeFineDelay.
7:6	Reserved.
5	**CkeSetup: CKE setup time**. Read-write. Reset: 0. This bit selects the default setup time for the CKE pins versus MEMCLK. 0=1/2 MEMCLK. 1=1 MEMCLK.
4:0	**CkeFineDelay: CKE fine delay**. Read-write. Cold reset: 0. This field specifies the time that the CKE pins are delayed from the default setup time. Bits Definition 1Fh-0h <CkeFineDelay>/64 MEMCLK delay

D18F2x9C_x0000_0[1:0]0[6:5] DRAM Read DQS Timing Control

Cold reset: 1E1E_1E1Eh. BIOS: See 2.9.3.7 [DRAM Training]. These registers delay the timing of read (input) DQS signals with respect to data.

Table 56: Index addresses for D18F2x9C_x0000_0[1:0]0[6:5]

D18F2x98[31:16]	D18F2x98[15:0]			
	0106h	0105h	0006h	0005h
0000h	DIMM 1 Bytes 7-4	DIMM 1 Bytes 3-0	DIMM 0 Bytes 7-4	DIMM 0 Bytes 3-0

Table 57: Byte lane mapping for D18F2x9C_x0000_0[1:0]0[6:5]

Register	Bits			
	29:25	21:17	13:9	5:1
D18F2x9C_x0000_0[1:0]05	Byte 3	Byte 2	Byte 1	Byte 0
D18F2x9C_x0000_0[1:0]06	Byte 7	Byte 6	Byte 5	Byte 4

Bits	Description
31:30	Reserved.
29:25	**RdDqsTime: read DQS timing control**. Read-write. See: D18F2x9C_x0000_0[1:0]0[6:5][5:1].
24:22	Reserved.
21:17	**RdDqsTime: read DQS timing control**. Read-write. See: D18F2x9C_x0000_0[1:0]0[6:5][5:1].
16:14	Reserved.
13:9	**RdDqsTime: read DQS timing control**. Read-write. See: D18F2x9C_x0000_0[1:0]0[6:5][5:1].
8:6	Reserved.

5:1	**RdDqsTime: read DQS timing control**. Read-write. Bits: Definition 1Fh-0h: <RdDqsTime>/64 MEMCLK delay
0	Reserved.

D18F2x9C_x0000_0008 DRAM Phy Control

Cold reset: 0208_0000h.

Table 58. DRAM Phy PLL Multiplier and Divide Ratio Values

MEMCLK Frequency(MHz)	PLL Multiplier	PLL Divide Ratio
200	16	4
400	16	2
533	32	3
667	40	3
800	32	2
933	56	3

Bits	Description
31:30	Reserved.
29:28	Reserved.
27:24	**PllDiv: phy PLL divider**. Read-only; updated-by-hardware. This field specifies the divide ratios that are encoded to the DDR phy PLL. Available encodings are as follows: Encoding / Divide Ratio / Encoding / Divide Ratio 0000b 1 0110b 256 0001b 2 0111b PLL is disabled 0010b 4 1000b 3 0011b 8 1001b 6 0100b 16 1111b-1010b Reserved 0101b 128
23:22	Reserved.
21:15	**PllMult: phy PLL multiplier**. Read-only; updated-by-hardware. This field specifies the multiplier values that are to be used for the DDR phy PLL. Bits: Multiplier 7Fh-00h: <PllMult>
14	Reserved.
13	**DqsRcvTrEn: DQS receiver training enable**. Read-write. BIOS: See 2.9.3.7.2. 1=Initiate hardware assisted read DQS receiver training. 0=Stop read DQS receiver training. This allows BIOS to reliably read the DQS receiver training data.
12	**WrLvOdtEn: write levelization ODT enabled**. Read-write. BIOS: See 2.9.3.7.1. 1=ODT specified by WrLvOdt is enabled during write levelization training. 0=ODT is disabled during write levelization training.

11:8	**WrLvOdt: write levelization ODT**. Read-write. BIOS: See 2.9.3.7.1. Specifies the state of the ODT pins when WrLvOdtEn=1. Bit[0] applies to ODT0; bit[1] applies to ODT1; etc. ODT numbers are as specified by D18F2x[4C:40]. For each bit, 1=ODT is enabled; 0=ODT is disabled. Tri-state enable for ODT is turned off by the phy while WrLvOdtEn=1.
7:6	**FenceTrSel: fence train select**. Read-write. BIOS: See 2.9.3.2.3. Specifies the flop to be used for phy based fence training. See PhyFenceTrEn. Bits Definition 00b PRE flop 01b RxDll flop 10b TxDll flop 11b TxPad flop
5	Reserved.
4	**TrDimmSel: training DIMM select**. Read-write. BIOS: See 2.9.3.7.1 and 2.9.3.7.2. Specifies which DIMM is to be trained. 0h=DIMM 0. 1h=DIMM 1. DIMM numbers are specified by Table 52.
3	**PhyFenceTrEn: phy fence training enable**. Read-write. BIOS: See 2.9.3.2.3. 1=Initiate phy based fence training. 0=Stop the phy based fence training engine.
2:1	Reserved.
0	**WrtLvTrEn: write levelization training enable**. Read-write. BIOS: See 2.9.3.7.1. 1=Initiate write levelization (tDQSS margining) training. 0=Stop driving DQS and exit write levelization training.

D18F2x9C_x0000_000B DRAM Phy Status

Reset: 0000_0000h. RAZ; write.

Bits	Description
31	**DynModeChange: dynamic mode change**. BIOS: See 2.9.3.8. 1=Phy enters the state specified by PhySelfRefreshMode.
30:24	Reserved.
23	**PhySelfRefreshMode: phy self refresh mode**. 1=Enter self refresh mode. 0=Exit self refresh mode. See DynModeChange.
22:0	Reserved.

D18F2x9C_x0000_000C DRAM Phy Miscellaneous

Cold reset: 4E73_0000h. BIOS: See 2.9.3.2.3 [Phy Fence Programming].

This register provides access to the DDR phy to control signal tri-state functionality. Based on the system configuration, BIOS may tri-state signals with associated chip selects that are unpopulated in an effort to conserve power. See Table 52 for processor pin map. This register also provides access to the DDR phy fence logic used to adjust the phase relationship between the data FIFO and the data going to the pad.

Bits	Description
31	Reserved.
30:26	**FenceThresholdTxDll: phy fence threshold transmit DLL**. Read-write. This field specifies the fence delay threshold value used for DQS receiver valid. This field is only used during DQS receiver enable training. See FenceThresholdTxPad.

25:21	**FenceThresholdRxDll: phy fence threshold DQS receiver enable**. Read-write. This field specifies the fence delay threshold value used for DQS receiver enable. See FenceThresholdTxPad.
20:16	**FenceThresholdTxPad: phy fence threshold transmit pad**. Read-write. This field specifies the fence delay threshold value used for write data, write DQS, Addr/Cmd, CS, ODT, and CKE. Bits Definition 1Fh-0h <FenceThresholdTxPad>/64 MEMCLK delay
15:14	Reserved.
13:12	**CKETri: CKE tri-state**. Read-write. CKETri has no effect; the CKE signals are never tri-stated.
11:8	**ODTTri: ODT tri-state**. Read-write. BIOS: See 2.9.3.8. 0=The ODT signals are not tri-stated unless directed by the DCT. 1=Tri-state ODT signals from the processor. This should only be set for unconnected ODT signals. The bits ODTTri[3:0] are mapped to package pins as follows: Bit Package pin name [0] MA_ODT[0] [1] MA_ODT[1] [2] MA_ODT[2] [3] MA_ODT[3]
7:0	**ChipSelTri: chip select tri-state**. Read-write. BIOS: See 2.9.3.8. 0=The chip select signals are not tri-stated unless directed to by the DCT. 1=Tri-state chip selects from the processor. This should only be set for unconnected chip select signals when motherboard termination is available. The bits ChipSelTri[7:0] are mapped to package pins as follows: Bit Package pin name [0] MA_CS_L[0] [1] MA_CS_L[1] [2] MA_CS_L[2] [3] MA_CS_L[3] [7:4] Unused pin

D18F2x9C_x0000_000D [DRAM Phy DLL Control

Cold reset: 0000_0000h. This register defines programmable options for the phy's DLLs for power savings. There are two identical sets of configuration registers: one for the transmit DLLs (those running off of the phy's internal PCLK which is running at rate of 2*MEMCLK) and receive DLLs (those running off of the DQS from the DIMMs).

Bits	Description
31:26	Reserved.
25:24	**RxDLLWakeupTime: receive DLL wakeup time**. Read-write. BIOS: See 2.9.3.8. This field specifies the number of PCLKs that the DLL standby signal must de-assert prior to a DLL relock event or before read traffic is sent to the receive DLLs.
23	Reserved.
22:20	**RxCPUpdPeriod: receive charge pump period**. Read-write. BIOS: See 2.9.3.8. This field specifies the number of DLL relocks required to keep the receive DLLs locked for the period where there is no read traffic.
19:16	**RxMaxDurDllNoLock: receive maximum duration DLL no lock**. Read-write. BIOS: See 2.9.3.5. This field specifies the number of PCLK cycles that occur before the phy DLLs relock. A DLL relock occurs every 2^RxMaxDurDllNoLock if there are no reads during the period. 0=DLL power saving disabled.

15:10	Reserved.
9:8	**TxDLLWakeupTime: transmit DLL wakeup time**. Read-write. BIOS: See 2.9.3.8. This field specifies the number of PCLK's that the DLL standby signal must de-assert prior to a DLL relock event or before write traffic is sent to transmit DLLs.
7	Reserved.
6:4	**TxCPUpdPeriod: transmit charge pump DLL wakeup time**. Read-write. BIOS: See 2.9.3.8. This specifies the number of DLL relocks required to keep the TxDLLs locked for the period where there is no write traffic.
3:0	**TxMaxDurDllNoLock: transmit maximum duration DLL no lock**. Read-write. BIOS: See 2.9.3.5. This field specifies the number of PCLK cycles that occur before the phy DLLs relock. A DLL relock occurs every 2^TxMaxDurDllNoLock if there are no writes during the period. 0=DLL power saving disabled.

D18F2x9C_x0000_00[24:10] DRAM DQS Receiver Enable Timing Control

BIOS: See 2.9.3.7 [DRAM Training]. Each of these registers control the timing of the receiver enable from the start of the read preamble with respect to MEMCLK. Each control includes a gross timing field and a fine timing field, the sum of which is the total delay.

Table 59: Index addresses for D18F2x9C_x0000_00[24:10]

D18F2x98[31:4]	D18F2x98[3:0]			
	4h	3h	1h	0h
000_0001h	DIMM 1 Bytes 3-2	DIMM 1 Bytes 1-0	DIMM 0 Bytes 3-2	DIMM 0 Bytes 1-0
000_0002h	DIMM 1 Bytes 7-6	DIMM 1 Bytes 5-4	DIMM 0 Bytes 7-6	DIMM 0 Bytes 5-4

Table 60: Byte lane mapping for D18F2x9C_x0000_00[24:10]

Register	Bits	
	24:16	8:0
D18F2x9C_x0000_001[0,3]	Byte 1	Byte 0
D18F2x9C_x0000_001[1,4]	Byte 3	Byte 2
D18F2x9C_x0000_002[0,3]	Byte 5	Byte 4
D18F2x9C_x0000_002[1,4]	Byte 7	Byte 6

Bits	Description
31:25	Reserved.
24:21	**DqsRcvEnGrossDelay: DQS receiver enable gross delay**. Read-write. Reset: 0001b. See: D18F2x9C_x0000_00[24:10][8:5].
20:16	**DqsRcvEnFineDelay: DQS receiver enable fine delay**. Read-write. Cold reset: 0. See: D18F2x9C_x0000_00[24:10][4:0].
15:9	Reserved.

8:5	**DqsRcvEnGrossDelay: DQS receiver enable gross delay**. Read-write. Reset: 0001b. Bits Description Ch-0h <DqsRcvEnGrossDelay*0.5> MEMCLK delay Fh-Dh Reserved
4:0	**DqsRcvEnFineDelay: DQS receiver enable fine delay**. Read-write. Cold reset: 0. Bits Definition 1Fh-0h <DqsRcvEnFineDelay>/64 MEMCLK delay

D18F2x9C_x0000_00[44:30] DRAM DQS Write Timing Control

BIOS: See 2.9.3.7 [DRAM Training]. These registers control the timing of write DQS with respect to MEMCLK. The delay starts at the rise edge of MEMCLK corresponding to the CAS-write-latency. The total delay for each byte is the sum of WrDqsGrossDly and WrDqsFineDly.

Table 61: Index addresses for D18F2x9C_x0000_00[44:30]

D18F2x98[31:4]	D18F2x98[3:0]			
	4h	3h	1h	0h
000_0003h	DIMM 1 Bytes 3-2	DIMM 1 Bytes 1-0	DIMM 0 Bytes 3-2	DIMM 0 Bytes 1-0
000_0004h	DIMM 1 Bytes 7-6	DIMM 1 Bytes 5-4	DIMM 0 Bytes 7-6	DIMM 0 Bytes 5-4

Table 62: Byte lane mapping for D18F2x9C_x0000_00[44:30]

Register	Bits	
	23:16	7:0
D18F2x9C_x0000_003[0,3]	Byte 1	Byte 0
D18F2x9C_x0000_003[1,4]	Byte 3	Byte 2
D18F2x9C_x0000_004[0,3]	Byte 5	Byte 4
D18F2x9C_x0000_004[1,4]	Byte 7	Byte 6

Bits	Description
31:24	Reserved.
23:21	**WrDqsGrossDly: DQS write gross delay**. Read-write. Reset: 0. See: D18F2x9C_x0000_00[44:30][7:5].
20:16	**WrDqsFineDly: DQS write fine delay**. Read-write. Cold reset: 0. See: D18F2x9C_x0000_00[44:30][4:0].
15:8	Reserved.
7:5	**WrDqsGrossDly: DQS write gross delay**. Read-write. Reset: 0. Bits Definition 100b-000b <WrDqsGrossDly*0.5> MEMCLK delay 111b-101b Reserved
4:0	**WrDqsFineDly: DQS write fine delay**. Read-write. Cold reset: 0. Bits Definition 1Fh-0h <WrDqsFineDly>/64 MEMCLK delay

D18F2x9C_x0000_00[51:50] DRAM Phase Recovery Control

BIOS: See 2.9.3.7 [DRAM Training]. These registers are used for hardware assisted DRAM training. Writes to these registers seed the phase recovery engine prior to training. Reads from the registers indicate how much the phase recovery engine has advanced to align the MEMCLK and DQS edges and is under hardware control. The total delay for each byte is the sum of PhRecGrossDlyByte and PhRecFineDlyByte, ranging from 0 to 1 and 63/64 MEMCLKs.

Table 63: Index addresses for D18F2x9C_x0000_00[51:50]

D18F2x98[31:8]	D18F2x98[7:0]	
	51h	50h
00_0000h	Bytes 7-4	Bytes 3-0

Table 64: Byte lane mapping for D18F2x9C_x0000_00[51:50]

Register	Bits			
	30:24	22:16	14:8	6:0
D18F2x9C_x0000_0050	Byte 3	Byte 2	Byte 1	Byte 0
D18F2x9C_x0000_0051	Byte 7	Byte 6	Byte 5	Byte 4

Bits	Description
31	Reserved.
30:29	**PhRecGrossDly: phase recovery gross delay**. Read-write; updated-by-hardware. Reset: X. See: D18F2x9C_x0000_00[51:50][6:5].
28:24	**PhRecFineDly: phase recovery fine delay**. Read-write; updated-by-hardware. Reset: X. See: D18F2x9C_x0000_00[51:50][4:0].
23	Reserved.
22:21	**PhRecGrossDly: phase recovery gross delay byte**. Read-write; updated-by-hardware. Reset: X. See: D18F2x9C_x0000_00[51:50][6:5].
20:16	**PhRecFineDly: phase recovery fine delay byte**. Read-write; updated-by-hardware. Reset: X. See: D18F2x9C_x0000_00[51:50][4:0].
15	Reserved.
14:13	**PhRecGrossDly: phase recovery gross delay**. Read-write; updated-by-hardware. Reset: X. See: D18F2x9C_x0000_00[51:50][6:5].
12:8	**PhRecFineDly: phase recovery fine delay**. Read-write; updated-by-hardware. Reset: X. See: D18F2x9C_x0000_00[51:50][4:0].
7	Reserved.
6:5	**PhRecGrossDly: phase recovery gross delay**. Read-write; updated-by-hardware. Reset: X. Bits Definition 11b-00b <PhRecGrossDly*0.5> MEMCLKs
4:0	**PhRecFineDly: phase recovery fine delay**. Read-write; updated-by-hardware. Reset: X. Bits Definition 1Fh-0h <PhRecFineDly>/64 MEMCLK delay

D18F2x9C_x0D0F_0[F,7:0]0[8,0] Data Byte Pad Driver Configuration

Cold reset: 0000_0033h. BIOS: Table 21 through Table 22. See 2.9.3.4.6 [DRAM Address Timing and Output Driver Compensation Control].

Table 65: Index addresses for D18F2x9C_x0D0F_0[F,7:0]0[8,0] pad group 0

D18F2x98[31:16]	D18F2x98[15:0]							
	0700h	0600h	0500h	0400h	0300h	0200h	0100h	0000h
0D0Fh	Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0

Table 66: Index addresses for D18F2x9C_x0D0F_0[F,7:0]0[8,0] pad group 2

D18F2x98[31:16]	D18F2x98[15:0]							
	0708h	0608h	0508h	0408h	0308h	0208h	0108h	0008h
0D0Fh	Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0

Table 67: Broadcast write index address for D18F2x9C_x0D0F_0[F,7:0]0[8,0]

D18F2x98[31:16]	D18F2x98[15:0]	
	0F08h	0F00h
0D0Fh	D18F2x9C_x0D0F_0[7:0]08	D18F2x9C_x0D0F_0[7:0]00

Bits	Description
31:7	Reserved.
6:4	**ProcOdtDqOvrd: processor on-die termination DQ override**. Read-write. This field specifies the resistance of the on-die termination resistors for the DQ pins. Writes to this field are only required if the DQ and DQS pins require different on-die termination values. This field is valid only when D18F2x94[ProcOdtDis]=0. Writes to D18F2x9C_x0000_0000[ProcOdt] will be written to this field. Bits — Definition 000b — 240 ohms +/- 20% 001b — 120 ohms +/- 20% 010b — 80 ohms +/- 20% 011b — 60 ohms +/- 20% 111b-100b — Reserved.
3:0	Reserved.

D18F2x9C_x0D0F_0[F,7:0]02 Data Byte Transmit Pre-Driver Calibration

Cold reset: xxxx_xxxxh. BIOS: See 2.9.3.2.4.

Table 68: Index addresses for D18F2x9C_x0D0F_0[F,7:0]02

D18F2x98[31:16]	D18F2x98[15:0]							
	0702h	0602h	0502h	0402h	0302h	0202h	0102h	0002h
0D0Fh	Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0

Table 69: Broadcast write index address for D18F2x9C_x0D0F_0[F,7:0]02

D18F2x98[31:16]	D18F2x98[15:0]
	0F02h
0D0Fh	D18F2x9C_x0D0F_0[7:0]02

Table 70: Valid values for D18F2x9C_x0D0F_0[F,7:0]02**[TxPreP, TxPreN]**

Bits	Description
0h	Slew Rate 0 (slowest)
8h-1h	Reserved
9h	Slew Rate 1
11h-Ah	Reserved
12h	Slew Rate 2
1Ah-13h	Reserved
1Bh	Slew Rate 3
23h-1Ch	Reserved
24h	Slew Rate 4
2Ch-25h	Reserved
2Dh	Slew Rate 5
35h-2Eh	Reserved
36h	Slew Rate 6
3Eh-37h	Reserved
3Fh	Slew Rate 7 (fastest)

Bits	Description
31:16	Reserved.
15	**ValidTxAndPre: pre-driver calibration code valid**. Read-write; cleared-when-done. 1=Copy the pre-driver calibration codes from this register and D18F2x9C_x0D0F_0[F,7:0]0[A,6] into the associated transmit pad. Hardware clears this field after the copy is complete.
14:12	Reserved.
11:6	**TxPreP: PMOS pre-driver calibration code**. Read-write. Specifies the rising edge slew rate of the transmit pad. See: Table 70. After updating this value, BIOS must program ValidTxAndPre=1 for the change to take effect.
5:0	**TxPreN: NMOS pre-driver calibration code**. Read-write. Specifies the falling edge slew rate of the transmit pad. See: Table 70. After updating this value, BIOS must program ValidTxAndPre=1 for the change to take effect.

D18F2x9C_x0D0F_0[F,7:0]0[A,6] Data Byte Transmit Pre-Driver Calibration 2

Cold reset: xxxx_xxxxh. BIOS: See 2.9.3.2.4.

Table 71: Index addresses for D18F2x9C_x0D0F_0[F,7:0]0[A,6] **pad group 1**

D18F2x98[31:16]	D18F2x98[15:0]							
	0706h	0606h	0506h	0406h	0306h	0206h	0106h	0006h
0D0Fh	Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0

Table 72: Index addresses for D18F2x9C_x0D0F_0[F,7:0]0[A,6] **pad group 2**

D18F2x98[31:16]	D18F2x98[15:0]							
	070Ah	060Ah	050Ah	040Ah	030Ah	020Ah	010Ah	000Ah
0D0Fh	Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0

Table 73: Broadcast write index address for D18F2x9C_x0D0F_0[F,7:0]0[A,6]

D18F2x98[31:16]	D18F2x98[15:0]	
	0F0Ah	0F06h
0D0Fh	D18F2x9C_x0D0F_0[7:0]0A	D18F2x9C_x0D0F_0[7:0]06

Bits	Description
31:12	Reserved.
11:6	**TxPreP: PMOS pre-driver calibration code**. Read-write. Specifies the rising edge slew rate of the transmit pad. See: Table 70 [Valid values for D18F2x9C_x0D0F_0[F,7:0]02[TxPreP, TxPreN]]. After updating this value, BIOS must program D18F2x9C_x0D0F_0[F,7:0]02[ValidTxAndPre]=1 for the change to take effect.
5:0	**TxPreN: NMOS pre-driver calibration code**. Read-write. Specifies the falling edge slew rate of the transmit pad. See: Table 70 [Valid values for D18F2x9C_x0D0F_0[F,7:0]02[TxPreP, TxPreN]]. After updating this value, BIOS must program D18F2x9C_x0D0F_0[F,7:0]02[ValidTxAndPre]=1 for the change to take effect.

D18F2x9C_x0D0F_0[F,7:0]0F Data Byte DLL Clock Enable

Cold reset: 0000_0013h.

Table 74: Index addresses for D18F2x9C_x0D0F_0[F,7:0]0F

D18F2x98[31:16]	D18F2x98[15:0]							
	070Fh	060Fh	050Fh	040Fh	030Fh	020Fh	010Fh	000Fh
0D0Fh	Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0

Table 75: Broadcast write index address for D18F2x9C_x0D0F_0[F,7:0]0F

D18F2x98[31:16]	D18F2x98[15:0]
	0F0Fh
0D0Fh	D18F2x9C_x0D0F_0[7:0]0F

Bits	Description
31:15	Reserved.
14:12	**AlwaysEnDllClks: always enable DLL clocks**. Read-write. BIOS: See 2.9.3.2.3. 0=DLL clocks are turned off during periods of inactivity. 1=DLL clocks remain on during inactivity. Prior to programming AlwaysEnDllClks to a value other than 000b, D18F2x9C_x0000_000D[RxMaxDurDllNoLock, TxMaxDurDllNoLock] must both be programmed to 0000b. The bits AlwaysEnDllClks[2:0] are mapped to DLLs as follows: Bit Definition 0 RxEn DLL 1 TxDq DLL 2 TxDqs DLL
11:0	Reserved.

D18F2x9C_x0D0F_0[F,7:0]10 Data Byte DLL Power Management

Table 76: Index addresses for D18F2x9C_x0D0F_0[F,7:0]10

D18F2x98[31:16]	D18F2x98[15:0]							
	0710h	0610h	0510h	0410h	0310h	0210h	0110h	0010h
0D0Fh	Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0

Table 77: Broadcast write index address for D18F2x9C_x0D0F_0[F,7:0]10

D18F2x98[31:16]	D18F2x98[15:0]
	0F10h
0D0Fh	D18F2x9C_x0D0F_0[7:0]10

Bits	Description
31:13	Reserved.
12	**EnRxPadStandby: enable receiver pad standby**. Read-write. Cold reset: 0. BIOS: See 2.9.3.5.1=Phy will enable Receiver Standby Mode when it is not receiving data to save power.
11:0	Reserved.

D18F2x9C_x0D0F_0[F,7:0]13 Data Byte DLL Configuration

Table 78: Index addresses for D18F2x9C_x0D0F_0[F,7:0]13

D18F2x98[31:16]	D18F2x98[15:0]							
	0713h	0613h	0513h	0413h	0313h	0213h	0113h	0013h
0D0Fh	Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0

Table 79: Broadcast write index address for D18F2x9C_x0D0F_0[F,7:0]13

D18F2x98[31:16]	D18F2x98[15:0]
	0F13h
0D0Fh	D18F2x9C_x0D0F_0[7:0]13

Bits	Description
31:15	Reserved.
14	**ProcOdtAdv: ProcOdt advance**. Read-write. Cold reset: 1. BIOS: See 2.9.3.2.2.1=Start the digital control for the ProcOdt (receive termination) 1 PCLK early.
13:8	Reserved.
7	**RxDqsUDllPowerDown: receive DQS upper DLL power down**. Read-write. Cold reset: 0. BIOS: See 2.9.3.8. 1=Power down the upper receiver DQS DLL. BIOS should set this bit when x4 DIMMs are not present.
6:2	Reserved.
1	**DllDisEarlyU: DLL disable early upper**. Read-write. Cold reset: 0. BIOS: See 2.9.3.8. 1=Disable upper receiver DQS DLL early timing for power savings.
0	**DllDisEarlyL: DLL disable early lower**. Read-write. Cold reset: 0. BIOS: See 2.9.3.8. 1=Disable lower receiver DQS DLL early timing for power savings.

D18F2x9C_x0D0F_0[F,7:0]1F Data Byte Receiver Configuration

Cold reset: 0000_2002h.

Table 80: Index addresses for D18F2x9C_x0D0F_0[F,7:0]1F

D18F2x98[31:16]	D18F2x98[15:0]							
	071Fh	061Fh	051Fh	041Fh	031Fh	021Fh	011Fh	001Fh
0D0Fh	Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0

Table 81: Broadcast write index address for D18F2x9C_x0D0F_0[F,7:0]1F

D18F2x98[31:16]	D18F2x98[15:0]
	0F1Fh
0D0Fh	D18F2x9C_x0D0F_0[7:0]1F

Bits	Description
31:5	Reserved.

4:3	**RxVioLvl: receiver voltage level**. Read-write. BIOS: See 2.9.3.2.1. This field specifies the VDDIO_MEM_S voltage level. Bits Definition 00b 1.5 V 01b 1.35 V 10b Reserved 11b Reserved
2:0	Reserved.

D18F2x9C_x0D0F_0[F,7:0]30 Data Byte DLL Configuration and PowerDown

Table 82: Index addresses for D18F2x9C_x0D0F_0[F,7:0]30

D18F2x98[31:16]	D18F2x98[15:0]							
	0730h	0630h	0530h	0430h	0330h	0230h	0130h	0030h
0D0Fh	Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0

Table 83: Broadcast write index address for D18F2x9C_x0D0F_0[F,7:0]30

D18F2x98[31:16]	D18F2x98[15:0]
	0F30h
0D0Fh	D18F2x9C_x0D0F_0[7:0]30

Bits	Description
31:6	Reserved.
5	**PchgPdTxCClkGateDis: precharge power down TxCCLK gate disable**. Read-write. Cold reset: 0. BIOS: 0. 1=Disable gating of upstream TxCCLK during precharge power down.
4:0	Reserved.

D18F2x9C_x0D0F_0[F,7:0]31 Data Byte Fence2 Threshold

BIOS: See 2.9.3.2.3 [Phy Fence Programming].

Table 84: Index addresses for D18F2x9C_x0D0F_0[F,7:0]31

D18F2x98[31:16]	D18F2x98[15:0]							
	0731h	0631h	0531h	0431h	0331h	0231h	0131h	0031h
0D0Fh	Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0

Table 85: Broadcast write index address for D18F2x9C_x0D0F_0[F,7:0]31

D18F2x98[31:16]	D18F2x98[15:0]
	0F31h
0D0Fh	D18F2x9C_x0D0F_0[7:0]31

Bits	Description
31:15	Reserved.
14	**Fence2EnableRxDll: phy fence2 enable receive DLL**. Read-write. Cold reset: 0. 1=Enable the use of Fence2ThresholdRxDll for DQS receiver enable fence threshold. 0=Use D18F2x9C_x0000_000C[FenceThresholdRxDll].
13:10	**Fence2ThresholdRxDll: phy fence2 threshold DQS receiver enable**. Read-write. Cold reset: 0. If Fence2EnableRxDll=1**,** this field specifies the fence delay threshold value used for DQS receiver enable. See Fence2ThresholdTxPad.
9	**Fence2EnableTxDll: phy fence2 enable transmit DLL**. Read-write. Cold reset: 0. 1=Enable the use of Fence2ThresholdTxDll for transmit DLL fence threshold. 0=Use D18F2x9C_x0000_000C[FenceThresholdTxDll].
8:5	**Fence2ThresholdTxDll: phy fence2 threshold transmit DLL**. Read-write. Cold reset: 0. If Fence2EnableTxDll=1**,** this field specifies the fence delay threshold value used for DQS receiver valid. This field is only used during DQS receiver enable training. See Fence2ThresholdTxPad.
4	**Fence2EnableTxPad: fence2 enable transmit pad**. Read-write. Cold reset: 0. 1=Enable the use of Fence2ThresholdTxPad for transmit pad fence threshold. 0=Use D18F2x9C_x0000_000C[FenceThresholdTxPad].
3:0	**Fence2ThresholdTxPad: phy fence2 threshold transmit pad**. Read-write. Cold reset: 0. If Fence2EnableTxPad=1**,** this field specifies the fence delay threshold value used for write data and write DQS. Bits Definition Fh-0h <Fence2ThresholdTxPad>/64 MEMCLK delay

D18F2x9C_x0D0F_2[1:0]02 Clock Transmit Pre-Driver Calibration

Cold reset: xxxx_xxxxh. BIOS: See 2.9.3.2.4.

Table 86: Index address mapping for D18F2x9C_x0D0F_2[1:0]02

D18F2x98[31:0]	Function
0D0F_2002h	Clock 0 Pad Group 0
0D0F_2102h	Clock 1 Pad Group 0

Bits	Description
31:16	Reserved.
15	**ValidTxAndPre: pre-driver calibration code valid**. Read-write; cleared-when-done. 1=Copy the pre-driver calibration codes from this register into the associated transmit pad. Hardware clears this field after the copy is complete.
14:12	Reserved.
11:6	**TxPreP: PMOS pre-driver calibration code**. Read-write. Specifies the rising edge slew rate of the transmit pad. See: Table 70 [Valid values for D18F2x9C_x0D0F_0[F,7:0]02[TxPreP, TxPreN]]. After updating this value, BIOS must program ValidTxAndPre=1 for the change to take effect.
5:0	**TxPreN: NMOS pre-driver calibration code**. Read-write. Specifies the falling edge slew rate of the transmit pad. See: Table 70 [Valid values for D18F2x9C_x0D0F_0[F,7:0]02[TxPreP, TxPreN]]. After updating this value, BIOS must program ValidTxAndPre=1 for the change to take effect.

D18F2x9C_x0D0F_[C,8,2][1:0]1F Receiver Configuration

Cold reset: 0000_2000h.

Table 87: Index address mapping for D18F2x9C_x0D0F_[C,8,2][1:0]1F

D18F2x98[31:0]	Function
0D0F_201Fh	Clock 0
0D0F_211Fh	Clock 1
0D0F_801Fh	Cmd/Addr 0
0D0F_811Fh	Cmd/Addr 1
0D0F_C01Fh	Address

Bits	Description
31:5	Reserved.
4:3	**RxVioLvl: receiver voltage level**. Read-write. BIOS: See 2.9.3.2.1. This field specifies the VDDIO_MEM_S voltage level. Bits Definition 00b 1.5 V 01b 1.35 V 10b Reserved 11b Reserved
2:0	Reserved.

D18F2x9C_x0D0F_2[1:0]30 Clock Configuration and Power Down

D18F2x98[11:8]	MEMCLK disable
0h	D18F2x88[MemClkDis[1:0]]
1h	D18F2x88[MemClkDis[3:2]]

Bits	Description
31:5	Reserved.
4	**PwrDn: power down**. Read-write. Cold reset: 0. BIOS: See 2.9.3.8. 1=Turn off DLL circuitry. BIOS should set this bit if both lanes of the clock lane pair are not supported by the package.
3:0	Reserved.

D18F2x9C_x0D0F_[C,8,2][1:0]31 Fence2 Threshold

BIOS: See 2.9.3.2.3 [Phy Fence Programming].

Table 88: Index address mapping for D18F2x9C_x0D0F_[C,8,2][1:0]31

D18F2x98[31:0]	Function
0D0F_2031h	Clock 0
0D0F_2131h	Clock 1

Table 88: Index address mapping for D18F2x9C_x0D0F_[C,8,2][1:0]31

0D0F_8031h	Cmd/Addr 0
0D0F_8131h	Cmd/Addr 1
0D0F_C031h	Address

Bits	Description
31:5	Reserved.
4	**Fence2EnableTxPad: fence2 enable transmit pad**. Read-write. Cold reset: 0. 1=Enable the use of Fence2ThresholdTxPad for transmit pad fence threshold. 0=Use D18F2x9C_x0000_000C[Fence-ThresholdTxPad].
3:0	**Fence2ThresholdTxPad: phy fence2 threshold transmit pad**. Read-write. Cold reset: 0. If Fence2EnableTxPad=1**,** this field specifies the fence delay threshold value used for CLK, Addr/Cmd, CS, ODT, and CKE. Bits Definition Fh-0h <Fence2ThresholdTxPad>/64 MEMCLK delay

D18F2x9C_x0D0F_4009 Cmp Receiver Configuration

Cold reset: 0000_2000h.

Bits	Description
31:16	Reserved.
15:14	**CmpVioLvl: receiver voltage level**. Read-write. BIOS: See 2.9.3.2.1. This field specifies the VDDIO_MEM_S voltage level. Bits Definition 00b 1.5 V 01b 1.35 V 10b Reserved 11b Reserved
13:0	Reserved.

D18F2x9C_x0D0F_[C,8][1:0]02 Transmit Pre-Driver Calibration

Cold reset: xxxx_xxxxh. BIOS: See 2.9.3.2.4.

Table 89: Index address mapping for D18F2x9C_x0D0F_[C,8][1:0]02

D18F2x98[31:0]	Function
0D0F_8002h	Cmd/Addr 0 Pad Group 0
0D0F_8102h	Cmd/Addr 1 Pad Group 0
0D0F_C002h	Address Pad Group 0

Bits	Description
31:16	Reserved.
15	**ValidTxAndPre: pre-driver calibration code valid**. Read-write; cleared-when-done. 1=Copy the pre-driver calibration codes from this register and D18F2x9C_x0D0F_[C,8][1:0][12,0E,0A,06] into the associated transmit pad. Hardware clears this field after the copy is complete.

14:12	Reserved.
11:6	**TxPreP: PMOS pre-driver calibration code**. Read-write. Specifies the rising edge slew rate of the transmit pad. See: Table 70 [Valid values for D18F2x9C_x0D0F_0[F,7:0]02[TxPreP, TxPreN]]. After updating this value, BIOS must program ValidTxAndPre=1 for the change to take effect.
5:0	**TxPreN: NMOS pre-driver calibration code**. Read-write. Specifies the falling edge slew rate of the transmit pad. See: Table 70 [Valid values for D18F2x9C_x0D0F_0[F,7:0]02[TxPreP, TxPreN]]. After updating this value, BIOS must program ValidTxAndPre=1 for the change to take effect.

D18F2x9C_x0D0F_[C,8][1:0][12,0E,0A,06] Transmit Pre-Driver Calibration 2

Cold reset: xxxx_xxxxh. BIOS: See 2.9.3.2.4.

Table 90: Index address mapping for D18F2x9C_x0D0F_[C,8][1:0][12,0E,0A,06]

D18F2x98[31:0]	Function
0D0F_8006h	Cmd/Addr 0 Pad Group 1
0D0F_800Ah	Cmd/Addr 0 Pad Group 2
0D0F_8106h	Cmd/Addr 1 Pad Group 1
0D0F_810Ah	Cmd/Addr 1 Pad Group 2
0D0F_C006h	Address Pad Group 1
0D0F_C00Ah	Address Pad Group 2
0D0F_C00Eh	Address Pad Group 3
0D0F_C012h	Address Pad Group 4

Bits	Description
31:12	Reserved.
11:6	**TxPreP: PMOS pre-driver calibration code**. Read-write. Specifies the rising edge slew rate of the transmit pad. See: Table 70 [Valid values for D18F2x9C_x0D0F_0[F,7:0]02[TxPreP, TxPreN]]. After updating this value, BIOS must program D18F2x9C_x0D0F_[C,8][1:0]02[ValidTxAndPre]=1 for the change to take effect.
5:0	**TxPreN: NMOS pre-driver calibration code**. Read-write. Specifies the falling edge slew rate of the transmit pad. See: Table 70 [Valid values for D18F2x9C_x0D0F_0[F,7:0]02[TxPreP, TxPreN]]. After updating this value, BIOS must program D18F2x9C_x0D0F_[C,8][1:0]02[ValidTxAndPre]=1 for the change to take effect.

D18F2x9C_x0D0F_812F Addr/Cmd Tri-state Configuration

Cold reset: 0000_00A0h. BIOS: See 2.9.3.8.

Bits	Description
31:8	Reserved.
7	**Add16Tri: MEMADD[16] tri-state**. Read-write. This field specifies tri-state control for the memory address[16] signal. 1=Signal is tri-stated. 0=Signal is not tri-stated.
6	Reserved.
5	**Add17Tri: MEMADD[17] tri-state**. Read-write. This field specifies tri-state control for the memory address[17] signal. 1=Signal is tri-stated. 0=Signal is not tri-stated.

4:1	Reserved.
0	**PARTri: MEMPAR tri-state**. Read-write. This field specifies tri-state control for the memory parity signal. 1=Signal is tri-stated. 0=Signal is not tri-stated.

D18F2x9C_x0D0F_C000 CKE 2.0X Pad Configuration

Cold reset: 0000_0003h.

Bits	Description
31:9	Reserved.
8	**LowPowerDrvStrengthEn: low power drive strength enable**. Read-write. LowPower-DrvStrengthEn has no effect; the CKE drive strength is always 2.0x.
7:0	Reserved.

D18F2x9C_x0D0F_E003 Phy Calibration Configuration

Cold reset: 0000_0210h.

Bits	Description
31:15	Reserved.
14	**DisAutoComp: disable automatic compensation**. Read-write. BIOS: See 2.9.3 and 2.9.3.2.4. 1=Disable the compensation control state machine. 0=The phy automatic compensation engine is enabled.
13	**DisablePreDriverCal: disable pre-driver calibration**. Read-write. BIOS: See 2.9.3 and 2.9.3.2.4. 1=Disables hardware update of pre-driver calibration codes.
12:0	Reserved.

D18F2x9C_x0D0F_E006 Phy PLL Lock Time

Cold reset: 0000_0190h.

Bits	Description
31:16	Reserved.
15:0	**PllLockTime: PLL lock time**. Read-write. BIOS: See 2.9.3.2.2. This field specifies the number of 5 ns periods the phy waits for PLLs to lock during a frequency change.

D18F2x9C_x0D0F_E00A Phy Dynamic Power Mode

Cold reset: 0000_0000h.

Bits	Description
31:15	Reserved.
14	**SelCsrPllPdMode: mode**. Read-write. BIOS: 0. 1=CsrPhySrPllPdMode selects power down mode. 0=Reserved.

13:12	**CsrPhySrPllPdMode: CSR phy self refresh power down mode**. Read-write. BIOS: 00b. Selects the PLL power down mode during phy self refresh when SelCsrPllPdMode=1. Bits Definition 00b No PLL power down 01b Reserved 10b PLL regulator power down 11b Reserved
11:0	Reserved.

D18F2xA0 DRAM Controller Miscellaneous

Reset: 0000_0000h. See 2.9.1 [DCT Configuration Registers] for general programming information about DCT configuration registers.

Bits	Description
31:0	Reserved.

D18F2xA4 DRAM Controller Temperature Throttle

Reset: 0000_0000h.

Bits	Description
31:3	Reserved.
2:1	**ThrottleEn: DRAM throttle enable**. Read-write. Reset: 00b. BIOS: See 2.5.5.2 and 2.9.3.5.This field specifies the average utilization of the channel if the M_EVENT_L pin is asserted. Throttling is accomplished by reducing command issue bandwidth based on historical command throttle values (Precharge, Activate, and AutoPrecharge = 1, Read and Write = 4). Bits Definition 00b 100% (No throttling) 01b 50% (New commands not issued more frequently than 1 every 2 * value NCLKs.) 10b 25% (New commands not issued more frequently than 1 every 4 * value NCLKs.) 11b 12.5% (New commands not issued more frequently than 1 every 8 * value NCLKs.)
0	**DoubleTrefRateEn: double Tref rate enable**. Read-write. Reset: 0. 1=Tref forced to 3.9 us auto-refresh interval when the M_EVENT_L pin is asserted. See 2.5.5.2 [M_EVENT_L].

D18F2xA8 DRAM Controller Miscellaneous 2

See 2.9.1 [DCT Configuration Registers] for general programming information about DCT configuration registers.

Bits	Description
31:23	Reserved.

22:21	**DbeGskMemClkAlignMode: DBE gasket MEMCLK align mode**. Read-write. Cold reset: 0. BIOS: See 2.9.3 and 2.9.3.5. Specifies the method used to align DDR commands with MEMCLK. Bits Definition 00b Command shift. 01b Pointer shift. 10b Phy MEMCLK shift. 11b Reserved
20	**BankSwap: swap bank address**. Read-write. Reset: 0. BIOS: See 2.9.3.5. 1=Swap the DRAM bank address bits. If D18F2x114[DctSelBankSwap]==1 then normalized address bits 10:8 are swapped with bits 15:13 else normalized address bits 11:9 are swapped with bits 15:13. This swap happens before D18F2x94[BankSwizzleMode] is applied. 0=Do not swap the DRAM bank address bits.
19:0	Reserved.

D18F2xAC DRAM Controller Temperature Status

Cold reset: 0000_0000h.

Bits	Description
31:1	Reserved.
0	**MemTempHot: Memory temperature hot**. Read; set-by-hardware; write-1-to-clear. 1=The M_EVENT_L pin was asserted indicating the memory temperature exceeded the normal operating limit; the DCT may be throttling the interface to aid in cooling (see D18F2xA4). See 2.5.5.2 [M_EVENT_L].

D18F2xF0 DRAM Controller Extra Data Offset

Reset: 8000_0000h.

The DCT includes an array of registers called D18F2xF4_x[FFF:0], which are defined following D18F2xF4. D18F2xF0 and D18F2xF4 are used to access D18F2xF4_x[FFF:0]. The register number (i.e., the number that follows "_x" in the register mnemonic) is specified by D18F2xF0[DctOffset]. Access to these registers is accomplished as follows:

- Reads:
  - Write the register number to D18F2xF0[DctOffset] with D18F2xF0[DctAccessWrite]=0.
  - Read the register contents from D18F2xF4.

- Writes:
  - Write all 32 bits to the register data to D18F2xF4 (individual byte writes are not supported).
  - Write the register number to D18F2xF0[DctOffset] with D18F2xF0[DctAccessWrite]=1.

Bits	Description
31	**DctAccessDone: DRAM controller access done**. Read-only. 1=The access to one of the D18F2xF4_x[FFF:0] registers is complete. 0=The access is still in progress.
30	**DctAccessWrite: DRAM controller read/write select**. Read-write. 0=Read one of the D18F2xF4_x[FFF:0] registers. 1=Write one of the D18F2xF4_x[FFF:0] registers.
29:28	Reserved.
27:0	**DctOffset: DRAM controller offset**. Read-write.

D18F2xF4 DRAM Controller Extra Data Port

Reset: xxxx_xxxxh. See D18F2xF0.

Bits	Description
31:0	**DctExtDataPort**. Read-write.

D18F2xF4_x06 DCT Read Timing

Reset: 0000_0000h.

Bits	Description
31:12	Reserved.
11:8	**TwrrdSD: write-to-read timing same DIMM**. Read-write. BIOS: See 2.9.3.4.3 [Twrrd and TwrrdSD (Write-to-Read DIMM Termination Turn-around)]. This specifies the minimum number of cycles from the last clock of *virtual CAS* of the first write operation to the clock in which CAS is asserted for a following read operation to a different chip select on the same DIMM. Bits Definition Ah-0h <TwrrdSD+1> clock(s) Fh-Bh Reserved
7	**TrdrdScEn: read to read timing same chip select enable**. Read-write. BIOS: 0. 0=Two reads to the same chip select are issued optimally with a minimum of 0 wait states. 1=Two reads to same chip select are issued with commands separated as specified by the TrdrdSD field.
6:4	Reserved.
3:0	**TrdrdSD: read to read timing same DIMM**. Read-write. BIOS: See 2.9.3.4.1 [Trdrd and TrdrdSD (Read-to-Read Timing)]. This field specifies the minimum number of cycles from the last clock of virtual CAS of a first read-burst operation to the clock in which CAS is asserted for a following read-burst operation that is to a different chip select on the same DIMM. Bits Definition 8h-0h <TrdrdSD+2> clocks Fh-9h Reserved

D18F2xF4_x16 DCT Write Timing

Reset: 0000_0000h.

Bits	Description
31:4	Reserved.
3:0	**TwrwrSD: write to write timing same DIMM**. Read-write. BIOS: See 2.9.3.4.2 [Twrwr and TwrwrSD (Write-to-Write Timing)]. This field specifies the minimum number of cycles from the last clock of virtual CAS of the first write-burst operation to the clock in which CAS is asserted for a following write-burst operation that is to a different chip select on the same DIMM. Bits Definition 9h-0h <TwrwrSD+1> clock(s) Fh-Ah Reserved

D18F2xF4_x30 DCT Skip Numerator

Cold reset: 0000_0400h.

Bits	Description
31:13	Reserved.
12:0	**DbeGskFifoNumerator: FIFO skip numerator**. Read-write. BIOS: See 2.9.3.2.2 and 2.9.3.4.7. This field specifies the skip numerator input term to logic which synchronizes between NCLK and MEMCLK. This field must be re-programmed if NCLK or MEMCLK frequency change.

D18F2xF4_x31 DCT Skip Denominator

Cold reset: 0000_0200h.

Bits	Description
31:13	Reserved.
12:0	**DbeGskFifoDenominator: FIFO skip denominator**. Read-write. BIOS: See 2.9.3.2.2 and 2.9.3.4.7. This field specifies the skip denominator input term to logic which synchronizes between NCLK and MEMCLK. This field must be re-programmed if NCLK or MEMCLK frequency change.

D18F2xF4_x32 DCT Transmit FIFO Control

Reset: 0000_0202h.

Bits	Description
31:16	Reserved.
15	**DataTxFifoSchedDlyNegSlot1: slot1 data transmit FIFO schedule delay negative**. Read-write. See: DataTxFifoSchedDlyNegSlot0.
14:13	Reserved.
12:8	**DataTxFifoSchedDlySlot1: slot1 data transmit FIFO schedule delay**. Read-write. See: DataTxFifoSchedDlySlot0.
7	**DataTxFifoSchedDlyNegSlot0: slot0 data transmit FIFO schedule delay negative**. Read-write. BIOS: See 2.9.3.5 and 2.9.3.4.7. Specifies the direction of delay as specified by DataTxFifoSchedDlySlot0. 1=DCT will delay the pull of write data versus sending of write CAS. 0=DCT will delay write CAS versus pull of write data. This field must be re-programmed if NCLK or MEMCLK frequency changes.
6:5	Reserved.
4:0	**DataTxFifoSchedDlySlot0: slot0 data transmit FIFO schedule delay**. Read-write. BIOS: See 2.9.3.5 and 2.9.3.4.7. Specifies FIFO slot0 timing for pulling DRAM write data to send to the phy versus sending write CAS to the phy in order to avoid FIFO overflow conditions. If DataTxFifoSchedDlyNegSlot0=1, this field specifies NCLK cycles, else this field specifies MEM-CLK cycles. This field must be re-programmed if NCLK or MEMCLK frequency changes. Bits Definition 1Fh-00h <DataTxFifoSchedDlySlot0> clock cycle(s)

D18F2xF4_x40 DRAM Timing 0

Reset: 0000_0000h.

Bits	Description
31:30	Reserved.
29:24	**Trc: row cycle time**. Read-write. BIOS: See 2.9.3.3. Specifies the minimum time in memory clock cycles from an activate command to another activate command or an auto-refresh command, all to the same chip select bank. Bits Definition 03h-00h Reserved. 26h-04h <Trc+16> clocks 3Fh-27h Reserved
23:21	Reserved.
20:16	**Tras: row active strobe**. Read-write. BIOS: See 2.9.3.3. Specifies the minimum time in memory clock cycles from an activate command to a precharge command, both to the same chip select bank. Bits Definition 15h-00h <Tras+15> clocks 1Fh-16h Reserved
15:12	Reserved.
11:8	**Trp: row precharge time**. Read-write. BIOS: See 2.9.3.3. Specifies the minimum time in memory clock cycles from a precharge command to an activate command or auto-refresh command, both to the same bank. Bits Definition 9h-0h <Trp+5> clocks Fh-Ah Reserved
7:4	Reserved.
3:0	**Trcd: RAS to CAS delay**. Read-write. BIOS: See 2.9.3.3. Specifies the time in memory clock cycles from an activate command to a read/write command, both to the same bank. Bits Definition 9h-0h <Trcd+5> clocks Fh-Ah Reserved

D18F2xF4_x41 DRAM Timing 1

Reset: 0000_0000h.

Bits	Description
31:19	Reserved.
18:16	**Twtr: internal DRAM write to read command delay**. Read-write. BIOS: See 2.9.3.3. Specifies the minimum number of memory clock cycles from a write operation to a read operation, both to the same chip select. This is measured from the rising clock edge following the last non-masked data strobe of the write to the rising clock edge of the next read command. Bits Definition 100b-000b <Twtr+4> clocks 111b-101b Reserved
15:11	Reserved.

10:8	**Trrd: row to row delay (or RAS to RAS delay)**. Read-write. BIOS: See 2.9.3.3. Specifies the minimum time in memory clock cycles between activate commands to different chip select banks. Bits Definition 100b-000b <Trrd+4> clocks 111b-101b Reserved
7:3	Reserved.
2:0	**Trtp: read CAS to precharge time**. Read-write. BIOS: See 2.9.3.3. Specifies the earliest time in memory clock cycles a page can be closed after having been read. Satisfying this parameter ensures read data is not lost due to a premature precharge. Bits Definition 100b-000b <Trtp+4> clocks for burst length of 32 or 64 bytes 111b-101b Reserved

D18F2xF4_x83 DCT ODT Control

Reset: 0000_0000h. See 2.9.3.4.5 [DRAM ODT Control].

Bits	Description
31:15	Reserved.
14:12	**WrOdtOnDuration: write ODT on duration**. Read-write. BIOS: 110b. Specifies the number of memory clock cycles that DIMM ODT is asserted for writes. Bits Definition 000b 0 clocks (Don't assert ODT) 101b-001b Reserved 111b-110b <WrOdtOnDuration> clocks
11:9	Reserved.
8	**WrOdtTrnOnDly: Write ODT Turn On Delay**. Read-write. BIOS: 0b. Specifies the number of memory clock cycles that DIMM ODT assertion is delayed relative to a write CAS. Bits Definition 0b 0 clocks (ODT asserted with CAS) 1b 1 clocks
7	Reserved.
6:4	**RdOdtOnDuration: Read ODT On Duration**. Read-write. BIOS: 110b. Specifies the number of memory clock cycles that DIMM ODT is asserted for reads. Bits Definition 000b 0 clocks (Don't assert ODT) 101b-001b Reserved 111b-110b <RdOdtOnDuration> clocks
3	Reserved.
2:0	**RdOdtTrnOnDly: Read ODT Turn On Delay**. Read-write. BIOS: MAX(0, D18F2x88[Tcl] - D18F2x84[Tcwl]). Specifies the number of memory clock cycles that DIMM ODT assertion is delayed relative to read CAS. Bits Description 100b-000b <RdOdtTrnOnDly> clocks 111b-101b Reserved

D18F2xF4_x180 DCT ODT Control

Reset: 0000_0000h. BIOS: Table 20.

Bits	Description
31:28	Reserved.
27:24	**RdOdtPatCs3: read ODT pattern chip select 3**. Read-write. This field represents the state of ODT[3:0] pins when a read occurs to the specified chip select.
23:20	Reserved.
19:16	**RdOdtPatCs2: read ODT pattern chip select 2**. Read-write. This field represents the state of ODT[3:0] pins when a read occurs to the specified chip select.
15:12	Reserved.
11:8	**RdOdtPatCs1: read ODT pattern chip select 1**. Read-write. This field represents the state of ODT[3:0] pins when a read occurs to the specified chip select.
7:4	Reserved.
3:0	**RdOdtPatCs0: read ODT pattern chip select 0**. Read-write. This field represents the state of ODT[3:0] pins when a read occurs to the specified chip select.

D18F2xF4_x182 DCT ODT Control

Reset: 0000_0000h. BIOS: Table 20.

Bits	Description
31:28	Reserved.
27:24	**WrOdtPatCs3: write ODT pattern chip select 3**. Read-write. This field represents the state of ODT[3:0] pins when a write occurs to the specified chip select.
23:20	Reserved.
19:16	**WrOdtPatCs2: write ODT pattern chip select 2**. Read-write. This field represents the state of ODT[3:0] pins when a write occurs to the specified chip select.
15:12	Reserved.
11:8	**WrOdtPatCs1: write ODT pattern chip select 1**. Read-write. This field represents the state of ODT[3:0] pins when a write occurs to the specified chip select.
7:4	Reserved.
3:0	**WrOdtPatCs0: write ODT pattern chip select 0**. Read-write. This field represents the state of ODT[3:0] pins when a write occurs to the specified chip select.

D18F2xF4_x200 DCT Power Management

Reset: 0000_0002h.

Table 91: BIOS Recommendations **for** D18F2xF4_x200**[Txp]**

Condition	D18F2xF4_x200
DdrRate	Txp
800	3h
1066	4h

Table 92: BIOS Recommendations **for** D18F2xF4_x200**[Txpdll]**

Condition	D18F2xF4_x200
DdrRate	Txpdll
800	0h
1066	3h

Bits	Description
31:13	Reserved.
12:8	**Txpdll: exit precharge and DLL PD to command delay**. Read-write. BIOS: Table 92. Specifies the minimum time that the DCT waits to issue a command after exiting precharge power-down mode if the DLL was also disabled. Bits Definition 15h-00h <Txpdll+10> clocks 1Fh-16h Reserved
7:4	Reserved.
3:0	**Txp: exit precharge PD to command delay**. Read-write. BIOS: Table 91. Specifies the minimum time that the DCT waits to issue a command after exiting precharge power-down mode. Bits Definition 1h-0h Reserved 7h-2h <Txp> clocks Fh-8h Reserved

D18F2x110 DRAM Controller Select Low

Reset: 0000_0000h.

Bits	Description
31:11	Reserved.
10	**MemCleared: memory cleared**. Read-only. 1=Memory has been cleared since the last warm reset. This bit is set by MemClrInit. See MemClrInit below.
9	**MemClrBusy: memory clear busy**. Read-only. 1=Memory clear operation is in progress. Reads or writes to DRAM while the memory clear operation is in progress result in undefined behavior.
8	**DramEnable: DRAM enabled**. Read-only. 1=The DCT initialization is complete (see 2.9.3.6 [DRAM Device Initialization]) or the DCT has exited from self refresh (D18F2x90[ExitSelfRef] transitions from 1 to 0).
7:4	Reserved.

3	**MemClrInit: memory clear initialization**. IF (D18F2x118[C6DramLock] == 1) THEN Read-only. ELSE Read-write; cleared-by-hardware. ENDIF. 1=The processor writes 0's to all locations of system memory attached to the processor and sets the MemCleared bit. See D18F1x40 and D18F1xF0.
	• The status of the memory clear operation can be determined by reading the MemClrBusy and MemCleared bits. This command is ignored if MemClrBusy=1 when the command is received.
	• BIOS must program the following registers before setting MemClrInit:
	  • D18F1x40 [DRAM Base]
	  • D18F1xF0 [DRAM Hole Address]
	  • D18F2x80 [DRAM Bank Address Mapping]
	  • D18F2x[4C:40] [DRAM CS Base Address]
	  • D18F2x[64:60] [DRAM CS Mask]
	  • D18F2x110 [DRAM Controller Select Low]
	  • D18F2x114 [DRAM Controller Select High]
	DramEnable must be set before setting MemClrInit. The memory prefetcher (see D18F2x11C) must be disabled before memory clear initialization and then can be re-enabled when MemCleared=1.

2:0	Reserved.

D18F2x114 DRAM Controller Select High

Reset: 0000_0000h.

Bits	Description
31:10	Reserved.
9	**DctSelBankSwap: select DRAM bank swap address**. Read-write. BIOS: 1. See D18F2xA8[BankSwap].
8:0	Reserved.

D18F2x118 Memory Controller Configuration Low

This register indicates the priority of request types. Variable priority requests enter the northbridge as medium priority and are promoted to high priority if they have not been serviced in the time specified by MctVarPriCntLmt. This feature may be useful for isochronous IO traffic. If isochronous traffic is specified to be high priority, it may have an adverse effect on the bandwidth and performance of the devices associated with the other types of traffic. However, if isochronous traffic is specified as medium priority, the processor may not be able to meet the isochronous bandwidth and latency requirements. The variable priority allows the memory controller to optimize DRAM transactions until isochronous traffic reaches a time threshold and must be serviced more quickly.

If a write requires a read-modify-write, arbitration occurs separately for the read and the write and the read has the same priority level as the write. If the priority of the write is changed for a read-modify-write then the priority of the read is changed as well to maintain the same priority was the write.

Bits	Description
31:28	**MctVarPriCntLmt: variable priority time limit**. Read-write. Reset: 0h. Bits / Definition / Bits / Definition 0000b 80 ns 1000b 720 ns 0001b 160 ns 1001b 800 ns 0010b 240 ns 1010b 880 ns 0011b 320 ns 1011b 960 ns 0100b 400 ns 1100b 1040 ns 0101b 480 ns 1101b 1120 ns 0110b 560 ns 1110b 1200 ns 0111b 640 ns 1111b 1280 ns
27:20	Reserved.
19	**C6DramLock**. Write-1-only. Reset: 0. BIOS: See 2.5.3.2.9 and 2.9.6. 1=The following registers are read-only: • D18F1x44 • D18F1xF0 • D18F2x[4C:40] • D18F2x[64:60] • D18F2x80 • D18F2x110 • D18F2x114
18:12	Reserved.
11:10	**MctPriWr: default write priority**. Read-write. Reset: 01b. BIOS: 01b. See: MctPriCpuRd.
9:8	**MctPriDefault: default non-write priority**. Read-write. Reset: 00b. BIOS: 00b. See: MctPriCpuRd.
7:6	**MctPriHiWr: high-priority VC set write priority**. Read-write. Reset: 00b. BIOS: 10b. See: MctPriCpuRd.
5:4	**MctPriHiRd: high-priority VC set read priority**. Read-write. Reset: 10b. BIOS: 10b. See: MctPriCpuRd.
3:2	**MctPriCpuWr: CPU write priority**. Read-write. Reset: 01b. BIOS: 01b. See: MctPriCpuRd.
1:0	**MctPriCpuRd: CPU read priority**. Read-write. Reset: 00b. BIOS: 00b. Bits / Definition / Bits / Definition 00b Low 10b High 01b Medium 11b Variable

D18F2x11C Memory Controller Configuration High

The two main functions of this register are to control write bursting and memory prefetching.

Write bursting. DctWrLimit specifies how writes may be burst from the DCT to improve DRAM efficiency. Bursting writes improves DRAM efficiency by minimizing the read-to-write turnaround time and the interference that non-latency critical stores have on latency critical loads. When the number of writes in the DCT reaches the value specified in DctWrLimit, then they become eligible for scheduling. Once eligible for scheduling, the priority based reorder algorithm picks the optimal write to increase DRAM bandwidth.

Rules regarding write bursting:
- Write bursting mode only applies to low-priority writes. Medium and high priority writes are not withheld from the DCT for write bursting.

- If write bursting is enabled, writes stay in the DCT until the threshold specified by DctWrLimit is reached. Once the threshold is reached, all writes in DCT are converted to medium-priority. Low-priority and medium-priority reads are not eligible until all converted medium-priority writes are scheduled.
- Any write in the DCT that matches the address of a subsequent access is promoted to either medium priority or the priority of the subsequent access, whichever is higher.

Memory prefetching. The DRAM prefetcher detects stride patterns in the stream of requests and then, for predictable stride patterns, generates prefetch requests. A stride pattern is a pattern of requests through system memory that are the same number of cache lines apart. The prefetcher supports strides of -4 to +4 cache lines, which can include alternating patterns (e.g. +1, +2, +1, +2), and can prefetch two strides ahead depending on the confidence. The prefetcher tracks up to 8 stride patterns simultaneously. Each of these stride patterns has a confidence level associated with it that is modified by how many requests match the stride pattern and is used to determine whether to fetch two strides ahead. The prefetcher behaves according to the following rules for each request:

- A prefetch request may be generated only when a demand request is received. When a demand address matches a tracked address and its stride pattern, then:
  - The confidence level for that stride pattern is incremented if less than PrefConfSat.
  - At this time, if the confidence level < PrefConf, no prefetch request is generated. If the confidence level >= PrefConf, one prefetch request is generated which is two consecutive strides ahead in the pattern.
  - Before the prefetch request is issued to the DCT, the prefetch address is checked that it still falls within the same 4 KB page as the request. If it the prefetch address crosses into a different 4 KB page, then the prefetch is squashed and the stride pattern is deallocated.
- Each time a request is received within +/- 4 cache lines of the last recorded address in the pattern and does not match the current stride pattern, then the confidence level is decreased by one.
- Each request that is within the same 4 KB page but outside the -4 to +4 cache line range (including the exact same cache line) of the last requested cache line of all the stride patterns tracked is ignored.
- Each request that is not within the same 4 KB page as the last requested cache line of the stride patterns tracked initiates a new stride pattern by displacing one of the existing least-recently-used stride patterns.
  - The confidence level is initialized to 0 for all new stride patterns.

Bits	Description
31	Reserved.
30	**FlushWr: flush writes command**. Read; Write-1-only; cleared-when-done. Reset: 0. BIOS: 0. Setting this bit causes write bursting to be cancelled and all outstanding writes to be flushed to DRAM. This bit is cleared when all writes are flushed to DRAM.
29	**FlushWrOnStpGnt: flush writes on stop-grant**. Read-write. Reset: 0. BIOS: 0. 1=Causes write bursting to be cancelled and all outstanding writes to be flushed to DRAM when in the stop-grant state. This bit should be set to ensure writes are drained to DRAM before reset is asserted for the suspend-to-RAM state.
28:25	Reserved.
24:22	**PrefConf: prefetch two-ahead confidence**. Read-write. Reset: 011b. BIOS: 1h. Confidence level required to issue one prefetch which is 2 strides ahead. Steady state keeps 2 prefetches ahead. Bits — Definition 000b — Reserved 111b-001b — <PrefConf*2>
21:20	Reserved.

19:18	**PrefConfSat: prefetch confidence saturation**. Read-write. Reset: 00b. BIOS: 0h. Specifies the point at which prefetch confidence level saturates and stops incrementing. Bits: Definition 00b: 15 01b: 7 10b: 3 11b: Reserved
17:15	Reserved.
14	**PrefCpuRdSzDis: prefetch CPU sized read disable**. Read-write. Reset: 1. BIOS: 1. 1=Disable generating prefetch requests for sized read requests from the cores and prevents sized read requests from the cores from consuming prefetch data. This bit has no effect if PrefCpuDis=1.
13	Reserved.
12	**PrefCpuDis: prefetch CPU access disable**. Read-write. Reset: 1. BIOS: See 2.9.3.5. 1=Disables core requests from triggering prefetch requests and prevents core requests from consuming prefetch data.
11:7	Reserved.
6:2	**DctWrLimit: memory controller write-burst limit**. Read-write. Reset: 18h. BIOS: See 2.9.3.5. Specifies the number of low-priority writes held in the memory controller queue before they are burst into the DCT. Bits: Definition 11h-00h: Reserved 1Eh-12h: <32-DctWrLimit> 1Fh: Write bursting disabled
1:0	Reserved.

D18F2x1C0 DRAM Training Control

Reset: 0000_0000h. See 2.9.3.7.5 [DRAM Training Pattern Generation].

Bits	Description
31:24	Reserved.
23	**RdTrainGo: read training go**. Read-write; cleared-when-done. 1=Initiate the data transfer from the DRAM interface to the read training buffer. This bit is cleared by hardware when the transfer is complete. See 2.9.3.7.5 [DRAM Training Pattern Generation].
22	**RdDramTrainMode: DRAM read training mode**. Read-write. 1=Enable read training mode. When RdDramTrainMode=1, WrDramTrainMode must be 0.
21	**AltAddrEn: alternate address enable**. Read-write. 1=Enable alternative address mode. See 2.9.3.7.5.2 [Alternative Address Mode]. 0=DRAM address specified by {D18F2x1CC[TrainAddrPtr[39:38]], D18F2x1C8[TrainAddrPtr[37:6]]}.
20	**DramTrainPdbDis: DRAM training prefetch buffer disable**. Read-write. BIOS: See 2.9.3.5. 1=Disable the use of additional prefetch buffers for read continuous pattern generation. 0=Allow use of additional prefetch buffers. If this bit is 0, all prefetching must be disabled by setting D18F2x11C[PrefCpuRdSzDis, PrefCpuDis]=11b.
19:18	Reserved.

17:2	**TrainLength: length in cache lines**. Read-write. Specifies the number of cache lines transferred from the write training buffer to the DRAM interface or from the DRAM interface to the read training buffer. See 2.9.3.7.5 [DRAM Training Pattern Generation].
1	**WrTrainGo: write training go**. Read-write; cleared-when-done. 1=Initiate the data transfer from the write training buffer to the DRAM interface. This bit is cleared by hardware when the transfer is complete. See 2.9.3.7.5 [DRAM Training Pattern Generation].
0	**WrDramTrainMode: DRAM write training mode**. Read-write. 1=Enable write training mode. When WrDramTrainMode=1, RdDramTrainMode must be 0.

D18F2x1C8 DRAM Training Address Pointer Low

Reset: 0000_0000h.

Bits	Description
31:0	**TrainAddrPtr[37:6]: DRAM training address pointer bits[37:6]**. Read-write. Specifies the lower bits of the DRAM address pointer in DRAM training mode. See D18F2x1CC.

D18F2x1CC DRAM Training Address Pointer High

Reset: 0000_0000h.

Bits	Description
31:26	**AltAddr3PtrIt: DRAM training alternate address pointer 3 iterations**. Read-write. See AltAddr1PtrIt. Specifies the number of times to iterate using AltAddr3Ptr before switching to TrainAddrPtr.
25:24	**AltAddr3Ptr[39:38]: DRAM training alternate address pointer 3bits[39:38]**. Read-write. See AltAddr1Ptr[39:38].
23:18	**TrainAddrPtrIt: DRAM training address pointer iterations**. Read-write. Specifies the number of times to iterate using TrainAddrPtr before switching to AltAddr1Ptr. The number of cache lines transferred using TrainAddrPtr is TrainAddrPtrIt + 1. See 2.9.3.7.5.2 [Alternative Address Mode].
17:16	**TrainAddrPtr[39:38]: DRAM training address pointer bits[39:38]**. Read-write. Specifies the upper bits of the DRAM address pointer in DRAM training mode. See D18F2x1C8. • TrainAddrPtr[39:6]={TrainAddrPtr[39:38], D18F2x1C8[TrainAddrPtr[37:6]]}; See 2.9.3.7.5.1 [Continuous Pattern Generation] and 2.9.3.7.5.2 [Alternative Address Mode].
15:10	**AltAddr2PtrIt: DRAM training alternate address pointer 2 iterations**. Read-write. See AltAddr1PtrIt.
9:8	**AltAddr2Ptr[39:38]: DRAM training alternate address pointer 2 bits[39:38]**. Read-write. See AltAddr1Ptr[39:38].
7:2	**AltAddr1PtrIt: DRAM training alternate address pointer 1 iterations**. Read-write. Specifies the number of times to iterate using AltAddr1Ptr before switching to AltAddr2Ptr. The number of cache lines transferred using AltAddr1Ptr is AltAddr1PtrIt + 1. See 2.9.3.7.5.2 [Alternative Address Mode].
1:0	**AltAddr1Ptr[39:38]: DRAM training alternate address pointer 1 bits[39:38]**. Read-write. Specifies the upper bits of the DRAM first alternate address pointer. See D18F2x[1E0:1D8]. • AltAddr1Ptr[39:6]={AltAddr1Ptr[39:38], D18F2x[1E0:1D8][AltAddr1Ptr[37:6]]}; See 2.9.3.7.5.2 [Alternative Address Mode].

D18F2x1D0 DRAM Write Training Buffer Address

Cold reset: 0000_0000h.

Bits	Description
31:10	Reserved.
9:0	**WrTrainBufAddr: write training buffer address**. Read-write; updated-by-hardware. Specifies the training data start address in the 1024-doubleword write training buffer. It is incremented by hardware after a write to D18F2x1D4. BIOS must program this register prior to filling the buffer or setting D18F2x1C0[WrTrainGo]. BIOS must write the lower four bits to 0h to begin on a cache line boundary.

D18F2x1D4 DRAM Write Training Data

Write-only. Reset: 0000_0000h.

Bits	Description
31:0	**WrTrainBufDat: write training buffer data**. Writing to this register writes the next doubleword of the training pattern to the training write buffer and increments D18F2x1D0[WrTrainBufAddr]. See D18F2x1D0[WrTrainBufAddr].

D18F2x[1E0:1D8] DRAM Training Alternate Address Pointer Low

Reset: 0000_0000h.

Table 93: Address mapping for D18F2x[1E0:1D8]

Register	Function
D18F2x1D8	Alternate Address 1
D18F2x1DC	Alternate Address 2
D18F2x1E0	Alternate Address 3

Bits	Description
31:30	Reserved.
29:0	**AltAddrPtr[35:6]: DRAM training alternate address pointer bits[35:6]**. Read-write. If D18F2x1C0[AltAddrEn]=1 specifies the lower bits of the DRAM alternate address pointer in DRAM training mode. See D18F2x1CC.

D18F2x1E8 DRAM Training Status

Reset: 0000_0000h.

Bits	Description
31:16	Reserved.
15:8	**TrainCmpSts2: DRAM training compare status 2**. Read-only; updated-by-hardware. Contains the comparison results between write and read training data. The last beat of read data in all cache line transfers is ignored. The read data of beats 0 to 6 are compared against the write data of beats 1 to 7. 1=comparison result is a miscompare. TrainCmpSts2[0] contains result for byte lane 0, TrainCmpSts2[7] contains the result of byte lane 7. Hardware clears the field when D18F2x1C0[RdDramTrainMode] changes from 0 to 1.
7:0	**TrainCmpSts: DRAM training compare status**. Read-only; updated-by-hardware. Contains the comparison results between write and read training data. 1=comparison result is a miscompare. TrainCmpSts[0] contains result for byte lane 0, TrainCmpSts[7] contains the result of byte lane 7. Hardware clears the field when D18F2x1C0[RdDramTrainMode] changes from 0 to 1.

3.10 Device 18h Function 3 Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space.

D18F3x00 Device/Vendor ID

Reset: 1703_1022h.

Bits	Description
31:16	**DeviceID: device ID**. Read-only.
15:0	**VendorID: vendor ID**. Read-only.

D18F3x04 Status/Command

IF (D18F3xE8[SvmCapable]==1) THEN Reset: 0010_0000h. ELSE Reset: 0000_0000h. ENDIF.

Bits	Description
31:16	**Status**. Read-only.
15:0	**Command**. Read-only.

D18F3x08 Class Code/Revision ID

Reset: 0600_0000h.

Bits	Description
31:8	**ClassCode**. Read-only. Provides the host bridge class code as defined in the PCI specification.
7:0	**RevID: revision ID**. Read-only.

D18F3x0C Header Type

Bits	Description
31:0	**HeaderTypeReg**. Value: 0080_0000h. The header type field indicates that there are multiple functions present in this device.

D18F3x34 Capability Pointer

Bits	Description
31:8	Reserved.
7:0	**CapPtr**. Read-only. IF (D18F3xE8[SvmCapable]==1) THEN Reset: F0h. ELSE Reset: 00h. ENDIF. Specifies the configuration-space offset of the capabilities pointer.

D18F3x40 MCA NB Control

Reset: 0000_0000h.

The machine check registers are used to configure the Machine Check Architecture (MCA) functions of the NB hardware and to provide a method for the NB to report errors in a way compatible with MCA. All of the NB MCA registers, except D18F3x44 [MCA NB Configuration], are accessible through the MCA-defined MSR method, as well as through PCI configuration space.

D18F3x40 [MCA NB Control] enables MCA reporting of each error checked by the NB. The global MCA error enables must also be set through MSR0000_017B [Global Machine Check Exception Reporting Control (MCG_CTL)]. The error enables in this register only affect error reporting through MCA. Actions which the NB may take in addition to MCA reporting are enabled through D18F3x44 [MCA NB Configuration].

Correctable and uncorrectable errors are logged in D18F3x48 [MCA NB Status Low], D18F3x4C [MCA NB Status High], and D18F3x50 [MCA NB Address Low] as they occur, as specified by D18F3x4C[Over]. Uncorrectable errors immediately result in a Machine Check exception.

Bits	Description
31:26	Reserved.
25	**UsPwDatErrEn: upstream data error enable**. Read-write. 1=Enables MCA reporting of upstream posted writes in which the EP bit is set.
24:18	Reserved.
17	**CpPktDatEn: completion packet error reporting enable**. Read-write. 1=Enables MCA reporting of completion packets with the EP bit set.
16	**NbIntProtEn: northbridge internal bus protocol error reporting enable**. Read-write. 1=Enables MCA reporting of protocol errors detected on the northbridge internal bus.
15:14	Reserved.
13	**DevErrEn: DEV error reporting enable**. Read-write. 1=Enables MCA reporting of SVM DEV errors.

12	**WDTRptEn: watchdog timer error reporting enable**. Read-write. 1=Enables MCA reporting of watchdog timer errors. The watchdog timer checks for NB system accesses for which a response is expected but no response is received. See D18F3x44 [MCA NB Configuration] for information regarding configuration of the watchdog timer duration. This bit does not affect operation of the watchdog timer in terms of its ability to complete an access that would otherwise cause a system hang. This bit only affects whether such errors are reported through MCA.
11	**AtomicRMWEn: atomic read-modify-write error reporting enable**. Read-write. 1=Enables MCA reporting of atomic read-modify-write (RMW) commands received from a link. Atomic RMW commands are not supported. An atomic RMW command results in a link error response being generated back to the requesting IO device. The generation of the link error response is not affected by this bit.
10	Reserved.
9	**TgtAbortEn: target abort error reporting enable**. Read-write. 1=Enables MCA reporting of target aborts to a link. The NB returns an error response back to the requestor with any associated data all 1s independent of the state of this bit.
8	**MstrAbortEn: master abort error reporting enable**. Read-write. 1=Enables MCA reporting of master aborts to a link. The NB returns an error response back to the requestor with any associated data all 1s independent of the state of this bit.
7:6	Reserved.
5	**SyncFloodEn: sync flood error reporting enable**. Read-write. 1=Enables MCA reporting of sync flood errors.
4:0	Reserved.

D18F3x44 MCA NB Configuration

Bits	Description
31	**NbMcaLogEn: NB MCA log enable**. Read-write. Reset: 0. 1=Enables logging (but not reporting) of NB MCA errors even if MCA is not globally enabled.
30	Reserved.
29	**DisMstAbtCpuErrRsp: master abort CPU error response disable**. Read-write. Reset: 0. 1=Disables master abort reporting through the MCA error-reporting banks. Master abort errors do not cause a sync flood when this bit is set.
28	**DisTgtAbtCpuErrRsp: target abort CPU error response disable**. Read-write. Reset: 0. 1=Disables target abort reporting through the MCA error-reporting banks. Target abort errors do not cause a sync flood when this bit is set.

27	**NbMcaToMstCpuEn: machine check errors to master CPU only**. Read-write. Reset: 0. BIOS: 1. 1=MCA errors in CMP device are only reported to core 0, and the NB MCA registers in MSR space (MSR0000_0410, MSR0000_0411 and MSR0000_0412) are only accessible from core 0; reads of these MSRs from other cores return 0 and writes are ignored. This field does not affect PCI-defined configuration space accesses to these registers, which are accessible from all cores. See 3.1 [Register Descriptions and Mnemonics] for a description of MSR space and 3 [Registers] for PCI-defined configuration space. 0=MCA errors may be reported to the CPU that originated the request, if applicable and known. • When the CPU which originates a request is known, it is stored in D18F3x4C[ErrCPU], regardless of the setting of NbMcaToMstCpuEn. • If IO originated the request, then the error is reported to core 0, regardless of the setting of NbMcaToMstCpuEn.
26	**CorrMcaExcEn: correctable error MCA exception enable**. Read-write. Reset: 0. 1=Correctable errors that are enabled for checking and logging cause a machine check exception (reporting) in addition to being logged.
25	**DisPciCfgCpuErrRsp: PCI configuration CPU error response disable**. Read-write. Reset: 0. 1=Disables generation of an error response to the core on detection of a master abort, target abort, or data error condition, and disables logging and reporting through the MCA error-reporting banks for PCI configuration accesses. Also, for NB WDT errors on PCI configuration accesses, this prevents sending an error response to the core, but does not affect logging and reporting of the NB WDT error. See D18F3x180[DisPciCfgCpuMstAbtRsp], which applies only to master aborts.
24	**IoRdDatErrEn: IO read data error log enable**. Read-write. Reset: 0. 1=Enables logging and reporting of read data errors (link defined master aborts, target aborts, and data error) for data destined for IO devices. 0=Read data errors for transactions from IO devices are not logged by MCA, although error responses may still be generated to the requesting IO device.
23:22	Reserved.
21	**SyncOnAnyErrEn: sync flood on any error enable**. Read-write. Reset: 0. 1=Enables generating a sync flood on detection of any NB MCA error that is uncorrectable.
20	**SyncOnWDTEn: sync flood on watchdog timer error enable**. Read-write. Reset: 0. BIOS: 1. 1=Enables generating a sync flood on detection of a watchdog timer error.
19:14	Reserved.
13:12	**WDTBaseSel: watchdog timer time base select**. Read-write. Reset: 0. BIOS: 0. Selects the time base used by the watchdog timer. The counter selected by WDTCntSel determines the maximum count value in the time base selected by WDTBaseSel. Bits / Definition / Bits / Definition 00b / 1.31 ms / 10b / 80 ns 01b / 1.28 us / 11b / Reserved

11:9	**WDTCntSel[2:0]: watchdog timer count select bits[2:0]**. Read-write. Reset: 0. BIOS: 0. Selects the count used by the watchdog timer. WDTCntSel is a 4-bit field composed of {D18F3x180[WDTCntSel[3]], D18F3x44[WDTCntSel[2:0]]}. The counter selected by WDTCntSel determines the maximum count value in the time base selected by WDTBaseSel. WDTCntSel is encoded as: Bits Definition Bits Definition 0h 4095 6h 63 1h 2047 7h 31 2h 1023 8h 8191 3h 511 9h 16383 4h 255 Fh-Ah Reserved 5h 127 Because WDTCntSel is split between two registers, care must be taken when programming WDTCntSel to ensure that a reserved value is never used by the watchdog timer or undefined behavior could result.
8	**WDTDis: watchdog timer disable**. Read-write. Cold reset: 0. 1=Disables the watchdog timer. The watchdog timer is enabled by default and checks for NB system accesses for which a response is expected and where no response is received. If such a condition is detected the outstanding access is completed by generating an error response back to the requestor. An MCA error may also be generated if enabled in D18F3x40 [MCA NB Control].
7	**IoErrDis: IO error response disable**. Read-write. Reset: 0. 1=Disables setting either Error bit in link response packets to IO devices on detection of a target abort, master abort, or data error condition.
6	**CpuErrDis: CPU error response disable**. Read-write. Reset: 0. 1=Disables generation of a read data error response to the core on detection of a target abort, master abort or data error condition.
5	**IoMstAbortDis: IO master abort error response disable**. Read-write. Reset: 0. 1=Signals target abort instead of master abort in link response packets to IO devices on detection of a master abort error condition. When IoMstAbortDis and D18F3x180[MstAbtChgToNoErrs] are both set, D18F3x180[MstAbtChgToNoErrs] takes precedence.
4:3	Reserved.
2	Reserved.
1	**CpuRdDatErrEn: CPU read data error log enable**. Read-write. Reset: 0. 1=Enables logging and reporting of read data errors (master aborts, target aborts, and data error) for data destined for the CPU. This bit should be clear if read data error logging is enabled for the remaining error reporting blocks in the CPU. Logging the same error in more than one block may cause a single error event to be treated as a multiple error event and cause the CPU to enter shutdown.
0	Reserved.

D18F3x48 MCA NB Status Low

Cold reset: xxxx_xxxxh.

Bits	Description
31:21	Reserved.
20:16	**ErrorCodeExt: extended error code**. Read-write; updated-by-hardware. Logs the extended error code when an error is detected. See Table 95 for the ErrorCodeExt encodings.
15:0	**ErrorCode: error code**. Read-write; updated-by-hardware. Logs an error code when an error is detected. See Table 40, Table 41, Table 42, Table 43, Table 44 for the ErrorCode encodings.

The NB is capable of reporting the following errors:

Table 94: NB error descriptions

Error Type	Description	Control Bits (D18F3x40)
Sync Flood	Unrecoverable error condition.	SyncFloodEn
Mst Abort	Master abort seen as result of link operation. Reasons for this error include requests to non-existent addresses, and requesting extended addresses while extended mode disabled (see D18F0x68[CHtExtNodeEn]). The NB returns an error response back to the requestor with any associated data all 1s independent of the state of the control bit.	MstrAbortEn
Target Abort	Target abort seen as result of link operation. The NB returns an error response back to the requestor with any associated data all 1s independent of the state of the control bit.	TgtAbortEn
RMW Error	An atomic read-modify-write (RMW) command was received from an IO link. Atomic RMW commands are not supported. An atomic RMW command results in a link error response being generated back to the requesting IO device. The generation of the link error response is not affected by the control bit.	AtomicRMWEn
WDT Error	NB WDT timeout due to lack of progress. The NB WDT monitors transaction completions. A transaction that exceeds the programmed time limit reports errors via the MCA. The cause of error may be another device which failed to respond.	WDTRptEn
DEV Error	SVM DEV error detected.	DevErrEn
NB Protocol Error	A protocol error was detected on the northbridge internal bus.	NbIntProtEn
Link Data Error	Data error detected on link.	CpPktDatEn UsPwDatErrEn

Table 95: NB error signatures, part 1

Error Type	**20:16 Ext. Error**	**Error Code (see D18F3x48 for encoding)**					
		Type	**10:9 PP**	**8 T**	**7:4 RRRR**	**3:2 II/TT**	**1:0 LL**
Reserved.	0_0000	-	-	-	-	-	-
Reserved	0_0001	-	-	-	-	-	-
Sync Error	0_0010	BUS	OBS	0	GEN	GEN	LG
Mst Abort	0_0011	BUS	SRC/OBS	0	RD/WR	MEM/IO	LG
Tgt Abort	0_0100	BUS	SRC/OBS	0	RD/WR	MEM/IO	LG
Reserved	0_0101	-	-	-	-	-	-
RMW Error	0_0110	BUS	OBS	0	GEN	MEM/IO	LG
WDT Error	0_0111	BUS	GEN	1	GEN	GEN	LG
Reserved	0_1000	-	-	-	-	-	-
DEV Error	0_1001	BUS	SRC/OBS	0	RD/WR	MEM/IO	LG
Link Data Error	0_1010	BUS	SRC/OBS	0	RD/WR/DWR	MEM/IO	LG
NB Protocol Error	0_1011	BUS	OBS	0	GEN	GEN	LG

Table 95: NB error signatures, part 1

Error Type	20:16 Ext. Error	Error Code (see D18F3x48 for encoding)					
		Type	10:9 PP	8 T	7:4 RRRR	3:2 II/TT	1:0 LL
Reserved	1_1111 to 0_1100	-	-	-	-	-	-

Table 96: NB error signatures, part 2

Error Type	D18F3x4C settings			
	UC	AddrV	PCC	ErrCPU
Sync Error	1	0	1	-
Mst Abort	1	1	If CPU source	Y
Tgt Abort	1	1	If CPU source	Y
RMW Error	1	1	0	-
WDT Error	1	1	1	-
DEV Error	1	1	0	-
Link Data Error	1	1	0	-
NB Protocol Error	1	0	1	-

D18F3x4C MCA NB Status High

Cold reset: xxxx_xxxxh.

Software is normally only allowed to write 0's to this register to clear the fields so subsequent errors may be logged. See MSRC001_0015[McStatusWrEn]. This register may be accessed through MSR0000_0411 [NB Machine Check Status (MC4_STATUS)] as well.

Bits	Description
31	**Val: error valid**. Read-write; set-by-hardware. 1=This bit indicates that a valid error has been detected. This bit should be cleared to 0 by software after the register has been read.
30	**Over: error overflow**. Read-write; set-by-hardware. 1=The NB attempted to record a new error with Val already set; an overflow occurred and the new error was not written. When Over and UC are both set, critical error information may have been lost, and software should terminate system processing to prevent data corruption (see 2.16.2.4 [Handling Machine Check Exceptions]). For certain conditions, a new error seen while Val is set may cause Over to be set, regardless of error priority or whether information was lost. Therefore, if UC is not indicated, there is no need to terminate the system, as any lost information was not critical. • If the existing error is overwritten, Over is not set. • Table 37 describes the conditions under which a younger error overwrites an older error.
29	**UC: error uncorrected**. Read-write; set-by-hardware. 1=The error was not corrected by hardware.
28	**En: error enable**. Read-write; set-by-hardware. 1=The MCA error reporting is enabled for this error in the MCA Control register.
27	Reserved.

26	**AddrV: error address valid**. Read-write; set-by-hardware. 1=The address saved in the address register is the address where the error occurred.
25	**PCC: processor context corrupt**. Read-write; set-by-hardware. 1=The state of the processor may be corrupted by the error condition. Reliable restarting might not be possible.
24:5	Reserved.
4	**BusErr: bus error**. Read-write; set-by-hardware. 1=The error was is associated with a transaction to or from the root complex.
3:2	Reserved.
1:0	**ErrCPU: error associated with core N**. Read-write; set-by-hardware. This field indicates which core within the processor is associated with the error. ErrCPU[1] = Error associated with core 1. ErrCPU[0] = Error associated with core 0.

D18F3x50 MCA NB Address Low

Cold reset: xxxx_xxxxh. Read-write.

D18F3x50 [MCA NB Address Low] and D18F3x54 [MCA NB Address High] carry the address associated with a machine check error, other fields, or both.

IF (D18F3x48[ErrorCodeExt] == 07h) THEN

Bits	Description
31:2	**ErrorAddr[31:2]**. Error address.
1	**RspDispatched**. 1=Response from link was dispatched.
0	**ReqDispatched**. 1=Request was dispatched.

ELSE

Bits	Description
31:0	**ErrorAddr[31:0]**. Error address.

ENDIF.

D18F3x54 MCA NB Address High

Cold reset: xxxx_xxxxh. Read-write.

IF (D18F3x48[ErrorCodeExt] == 07h) THEN

Bits	Description
31:28	**WaitCode: RAQ wait code**. • [63]=1 means all inbound data has not been transferred. • [62]=1 means ordering rules not satisfied for Dispatch of Response. • [61]: Reserved. • [60]=1 means lack of downstream credits.
27	**Priority**. 1=High. 0=Low.

Bits	Description
26:24	**PktRequester: packet requester**. Bits: Requester 000b: Core 0 001b: Core 1 010b: Reserved 011b: Reserved 100b: IO device 101b: Northbridge 11xb: Reserved
23:22	**PktTarget: packet target**. Bits: Target 00b: DRAM 01b: Northbridge 10b: MMIO/IO/FCH 11b: Reserved
21:16	**LinkCmd: link command**.
15:8	Reserved.
7:0	**ErrorAddr[39:32]: error address**.

ELSE

Bits	Description
31:8	Reserved.
7:0	**ErrorAddr[39:32]: error address**.

ENDIF.

D18F3x64 Hardware Thermal Control (HTC)

See 2.10.3.1 [PROCHOT_L and Hardware Thermal Control (HTC)] for information on HTC. D18F3x64 is reserved if D18F3xE8[HtcCapable]=0.

Bits	Description
31	**HtcLock: HTC lock**. Read; write-1-only. Reset: 0. 1=HtcPstateLimit, HtcHystLmt, HtcTmpLmt, and HtcEn are read-only. 0=HtcPstateLimit, HtcHystLmt, HtcTmpLmt, and HtcEn are read-write.
30:28	**HtcPstateLimit: HTC P-state limit select**. Read-write. Reset: Product-specific. Specifies the P-state limit of all cores when in the HTC-active state. The HtcPstateLimit to apply is not changed if the value of this field is greater than MSRC001_0061[PstateMaxVal]. See 2.10.3.1 [PROCHOT_L and Hardware Thermal Control (HTC)].
27:24	**HtcHystLmt: HTC hysteresis**. Read-write. Reset: Product-specific. The processor exits the HTC-active state when the temperature selected by HtcSlewSel is less than the HTC temperature limit (HtcTmpLmt) minus the HTC hysteresis (HtcHystLmt). Bits: Description Fh-0h: <HtcHystLmt*0.5>
23	**HtcSlewSel: HTC slew-controlled temperature select**. Read-write. Reset: 0. 1=HTC logic is driven by the slew-controlled temperature, Tctl, specified in D18F3xA4[CurTmp]. 0=HTC logic is driven by the measured control temperature with no slew controls.

22:16	**HtcTmpLmt: HTC temperature limit**. Read-write. Reset: Product-specific. The processor enters the HTC-active state when the temperature selected by HtcSlewSel reaches or exceeds the temperature limit HtcTmpLmt. Bits: Description 7Fh-00h: <HtcTmpLmt*0.5 + 52>
15:8	Reserved.
7	**PslApicLoEn: P-state limit lower value change APIC interrupt enable**. Read-write. Reset: 0. PslApicLoEn and PslApicHiEn enable interrupts using APIC330 of each core when the active P-state limit in MSRC001_0061[CurPstateLimit] changes. PslApicLoEn enables the interrupt when the limit value becomes lower (indicating higher performance). PslApicHiEn enables the interrupt when the limit value becomes higher (indicating lower performance). 1=Enable interrupt.
6	**PslApicHiEn: P-state limit higher value change APIC interrupt enable**. Read-write. Reset: 0. See: PslApicLoEn.
5	**HtcActSts: HTC-active status**. Read; set-by-hardware; write-1-to-clear. Reset: 0. This bit is set by hardware when the processor enters the HTC-active state. It is cleared by writing a 1 to it.
4	**HtcAct: HTC-active state**. Read-only. Reset: X. 1=The processor is currently in the HTC-active state. 0=The processor is not in the HTC-active state.
3:1	Reserved.
0	**HtcEn: HTC enable**. Read-write. Reset: 0. BIOS: IF (HtcTmpLmt == 00h) THEN 0. ELSE 1. ENDIF. 1=HTC is enabled; the processor is capable of entering the HTC-active state.

D18F3x6C Upstream Data Buffer Count

Buffer allocation requirements:

- UpHiNpreqDBC + UpHiPreqDBC + UpLoRespDBC + UpLoNpreqDBC + UpLoPreqDBC <=16.
- One buffer must be allocated for each enabled channel. The low priority channels are always enabled.
- Buffer allocations cannot be decreased through software. To decrement the buffers allocated to a channel generate a reset and then reassign buffers to the new settings.

Bits	Description
31:24	Reserved.
23:20	**UpHiNpreqDBC: upstream high priority non-posted request data buffer count**. Read-write. Reset: 0h. BIOS: Table 8.
19:16	**UpHiPreqDBC: upstream high priority posted request data buffer count**. Read-write. Reset: 0h. BIOS: Table 8.
15:12	Reserved.
11:8	**UpLoRespDBC: upstream low priority response data buffer count**. Read-write. Reset: 1h. BIOS: Table 8.
7:4	**UpLoNpreqDBC: upstream low priority non-posted request data buffer count**. Read-write. Reset: 1h. BIOS: Table 8.
3:0	**UpLoPreqDBC: upstream low priority posted request data buffer count**. Read-write. Reset: 1h. BIOS: Table 8.

D18F3x74 Upstream Command Buffer Count

Buffer allocation requirements:
- UpHiNpreqCBC + UpHiPreqCBC + UpLoNpreqCBC + UpLoPreqCBC <=16.
- UpLoRespCBC <=8.
- One buffer must be allocated for each enabled channel. The low priority channels are always enabled.
- Buffer allocations cannot be decreased through software. To decrement the buffers allocated to a channel generate a reset and then reassign buffers to the new settings.

Bits	Description
31:24	Reserved.
23:20	**UpHiNpreqCBC: upstream high priority non-posted command buffer count**. Read-write. Reset: 0. BIOS: Table 8.
19:16	**UpHiPreqCBC: upstream high priority posted command buffer count**. Read-write. Reset: 0. BIOS: Table 8.
15:12	Reserved.
11:8	**UpLoRespCBC: upstream low priority response command buffer count**. Read-write. Reset: 1h. BIOS: Table 8.
7:4	**UpLoNpreqCBC: upstream low priority non-posted command buffer count**. Read-write. Reset: 1h. BIOS: Table 8.
3:0	**UpLoPreqCBC: upstream low priority posted command buffer count**. Read-write. Reset: 1h. BIOS: Table 8.

D18F3x7C In-Flight Queue Buffer Allocation

Read-write. BIOS: See 2.9.3.5.

- Buffer allocation requirements:
  - FreePoolBC + D18F3x17C[HiPriNpBC] + D18F3x17C[HiPriPBC] + LoPriNpBC + LoPriPBC + CpuBC <= 28
  - One buffer must be allocated for each enabled channel. The CPU and low priority channels are always enabled.
  - Buffer allocations cannot be decreased through software. To decrement the buffers allocated to a channel generate a reset and then reassign buffers to the new settings.

Bits	Description
31:30	Reserved.
29:24	**FreePoolBC: free pool buffer count**. Reset: 1h. This field must be at least 1.
23:22	Reserved.
21:16	**LoPriNPBC: low priority channel non-posted buffer count**. Reset: 1h.
15:14	Reserved.
13:8	**LoPriPBC: low priority channel posted buffer count**. Reset: 1h.
7:6	Reserved.
5:0	**CpuBC: CPU buffer count**. Reset: 1h.

D18F3x80 ACPI Power State Control Low

Read-write. Reset: 0000_0000h. BIOS: 0000_0000h.

D18F3x80 and D18F3x84 consist of eight identical 8-bit registers, one for each System Management Action Field (SMAF) code associated with STPCLK assertion commands from the link. Refer to the table below for the associated ACPI state and SMAF code for each of the 8 registers. Some ACPI states and associated SMAF codes may not be supported in certain conditions. Refer to Table 7 for information on which states are supported. See D18F4x138 and D18F4x13C for the clock divisors that apply to each SMAF code.

Table 97: SMAF to ACPI state mapping

SMAF	ACPI State	Description
7	Reserved.	Reserved.
6	S4/S5	Initiated by a processor access to the ACPI-defined PM1_CNTa register.
5	Reserved.	Reserved.
4	S3	Initiated by a processor access to the ACPI-defined PM1_CNTa register.
3	Reserved.	Reserved.
2	Reserved.	Reserved.
1	Reserved.	Reserved.
0	Reserved.	Reserved.

Bits	Description
31:0	Reserved.

D18F3x84 ACPI Power State Control High

Reset: 0000_0000h. Read-write. BIOS: 0006_0006h.

See D18F3x80 for detail about the System Management Action Field (SMAF).

Bits	Description
31:19	Reserved.
18	**Smaf6DramMemClkTri: SMAF 6 DRAM memory clock tri-stated**. BIOS: 1. See: D18F3x84[Smaf4DramMemClkTri].
17	**Smaf6DramSr: SMAF 6 DRAM self-refresh**. BIOS: 1. See: D18F3x84[Smaf4DramSr].
16:3	Reserved.
2	**Smaf4DramMemClkTri: SMAF 4 DRAM memory clock tri-stated**. 1=MEMCLKs are tri-stated while in the low-power state. DramSr is required to be set if this bit is set. See 2.5.5.1 [DRAM Self-Refresh].
1	**Smaf4DramSr: SMAF 4 DRAM self-refresh**. 1=DRAM is enabled to be placed into self-refresh while in the low-power state. See 2.5.5.1 [DRAM Self-Refresh].
0	Reserved.

D18F3x88 NB Configuration Low

Reset: 0000_0200h. MSRC001_001F[31:0] is an alias of D18F3x88.

Bits	Description
31:0	Reserved.

D18F3x8C NB Configuration High

Reset: 0000_0000h. MSRC001_001F[63:32] is an alias of D18F3x8C.

Bits	Description
31:27	Reserved.
26	**EnConvertToNonIsoc**: **enable conversion to non-isochronous**. Read-write. BIOS: 1. 1=Convert peer-to-peer isochronous requests to non-isochronous requests (the Isoc bit in the downstream request packet is low); however, the Isoc bit in the downstream response to the requester is still set in such a case. In non-IFCM mode, the link-defined Isoc bit in the request packet is cleared as it is reflected downstream in a peer-to-peer access as well.
25:15	Reserved.
14	**EnableCf8ExtCfg: enable CF8 extended configuration cycles**. Read-write. 1=Allows the IO configuration space access method, IOCF8 and IOCFC, to be used to generate extended configuration cycles by enabling IOCF8[27:24].
13	**DisUsSysMgtReqToNcHt: disable upstream system management request to link**. Read-write. 1=Disables downstream reflection of upstream STPCLK and x86 legacy input system management commands (in order to work around potential deadlock scenarios related to reflection regions).
12:0	Reserved.

D18F3xA0 Power Control Miscellaneous

Bits	Description
31:28	Reserved.
27:16	**ConfigId: configuration identifier**. Read-only. Reset: Product-specific. Specifies the configuration ID associated with the product.
15:10	Reserved.
9	**SviHighFreqSel: SVI high frequency select**. Read-write. Reset: 0. BIOS: 1. 0=400 kHz. 1=3.4 MHz. Writes to this field take effect after the next SVI command.
8	Reserved.
7	**PsiVidEn: PSI_L bit VID enable**. Read-write. Cold reset: 0. BIOS: See 2.5.1.4.1 [PSI_L Bit]. This bit specifies how the PSI_L bit for VDDCR_CPU is controlled. 1=See D18F3xA0[PsiVid]. 0=The PSI_L bit is always 1.
6:0	**PsiVid: PSI_L bit VID threshold**. Read-write. Cold reset: 0. BIOS: See 2.5.1.4.1 [PSI_L Bit]. If D18F3xA0[PsiVidEn]==1, this field specifies a VID code that determines the state of the PSI_L bit for the VDDCR_CPU plane. Whenever the VID code generated by the processor for the VDDCR_CPU plane is less than (voltage is greater than) PsiVid, the PSI_L bit is sent as a 1. Whenever the VID code generated by the processor for the VDDCR_CPU plane is greater than or equal to (voltage less than or equal to) PsiVid, the PSI_L bit is sent as a 0. See 2.5.1.4.1 [PSI_L Bit].

D18F3xA4 Reported Temperature Control

The slew rate controls in this register are used to filter processor temperature measurements. Separate controls are provided for a measured temperature that is higher or lower than Tctl. The per-step timer counts as long as the measured temperature stays either above or below Tctl. Each time the measured temperature changes to the other side of Tctl, the step timer resets, and Tctl is not changed. If, for example, step times are enabled in both directions, Tctl=62.625, and the measured temperature keeps jumping quickly between 62.5 and 63.0, then (assuming the step times are long enough) Tctl would not change. However, once the measured temperature settles on one side of Tctl, Tctl can step toward the measured temperature. If the difference of measured temperature minus Tctl is greater than the value set by MaxTmpDiffUp, then Tctl is set equal to the measured temperature. See 2.10 [Thermal Functions].

Bits	Description
31:21	**CurTmp: current temperature**. Read-only. Reset: X. Specifies the current control temperature with the slew-rate controls applied. See 2.10.1 [The Tctl Temperature Scale]. Bits: Definition 7FFh-000h: <CurTmp*0.125>
20:13	Reserved.
12:8	**PerStepTimeDn: per 1/8th step time down**. Read-write. Cold reset: 18h. BIOS: 0Fh. This specifies the time per 1/8 step of Tctl when the measured temperature is less than the Tctl. It is encoded the same as PerStepTimeUp.
7	**TmpSlewDnEn: temperature slew downward enable**. Read-write. Cold reset: 0. BIOS: 1. 1=Slew rate controls in the downward direction are enabled. 0=Downward slewing disabled; if the measured temperature is detected to be less than Tctl then Tctl is updated to match the measured temperature.
6:5	**TmpMaxDiffUp: temperature maximum difference up**. Read-write. Cold reset: 00b. BIOS: 11b. This specifies the maximum difference between Tctl and the measured temperature, when the measured value is greater than Tctl (i.e., when the temperature has risen). If this difference exceeds the specified value, Tctl jumps to the measured temperature value. This field is encoded as follows: Bits: Definition 00b: Upward slewing disabled; if the measured temperature is detected to be greater than Tctl then Tctl is updated to match the measured temperature. 01b: Tctl is held to less than or equal to measured temperature minus 1.0. 10b: Tctl is held to less than or equal to measured temperature minus 3.0. 11b: Tctl is held to less than or equal to measured temperature minus 9.0.
4:0	**PerStepTimeUp: per 1/8th degree step time up**. Read-write. Cold reset: 00h. BIOS: 0Fh. This specifies the time per 1/8 step of Tctl when the measured temperature is greater than the reported temperature. It is encoded as follows: Bits: Definition 1Fh-00h: <(PerStepTimeUp[2:0] + 1) * 10^PerStepTimeUp[4:3]> ms, ranging from 1 ms to 8000 ms.

D18F3xD4 Clock Power/Timing Control 0

Bits	Description
31:19	Reserved.
18	Reserved.

17	**ClockGatingEnDram: clock gating enable DRAM**. Read-write. Reset: 0. BIOS: 1. Specifies whether NCLK gating to the DRAM controller is enabled. 1=Enabled. 0=Disabled. See 2.5.4.3 [NB Clock Gating].
16	**DisNclkGatingIdle: disable NCLK gating when idle**. Read-write. Reset: 0. 1=NCLK gating is disallowed when NCLK is ramped down. 0=NCLK gating is allowed when NCLK is ramped down. See 2.5.4.2 [NB Clock Ramping] and 2.5.4.3 [NB Clock Gating].
15:12	**NbOutHyst: Northbridge outbound hysteresis**. Read-write. Reset: 0. BIOS: 04h. Specifies the hysteresis time after the IFQ is emptied until the NB de-asserts the outbound wake signal. Bits Time 0h 0 Eh-1h <80 ns * 2^(NbOutHyst - 1)> Fh Wake de-assertion disabled See 2.5.4.3 [NB Clock Gating].
11:8	**ClkRampHystSel: clock ramp hysteresis select**. Read-write. Reset: 0. BIOS: Fh. Specifies the hysteresis time used when ramping up to service probe requests. Hysteresis time= 320 ns * (1 + ClkRampHystSel). See 2.5.3.2.5 [C-states and Probe Requests].
7:6	Reserved.
5:0	**MainPllOpFreqId: main PLL operating frequency ID**. Read-write; reset-applied. Cold reset: Product-specific. Specifies the COF of the main PLL. See MSRC001_0071[MainPllOpFreqIdMax] for the maximum supported frequency. • Main PLL COF = 100 MHz * (D18F3xD4[MainPllOpFreqId] + 10h). • D18F3xD4[MainPllOpFreqId] must be programmed as specified by MSRC001_0071[MainPllOpFreqIdMax].

D18F3xD8 Clock Power/Timing Control 1

Bits	Description
31:12	Reserved.
11:7	**ExtndTriDly: extend tri-state delay**. Read-write. Cold reset: 0_1111b. Specifies a delay in REFCLKs that the processor leaves the SVD signal tri-stated after receiving an ACK from the voltage regulator.

6:4	**VSRampSlamTime**. Read-write. Reset: 0. BIOS: Voltage Ramp Time[1]. Specifies the time the processor waits for voltage increases to complete before beginning an additional voltage change or a frequency change. See 2.5.1.5.1 [Hardware-Initiated Voltage Transitions]. Ramp time = (VSRampSlamTime / 12.5 mV) * ABS(destination voltage - current voltage). Bits / Time / Bits / Time 000b 6.25 us 100b 2.50 us 001b 5.00 us 101b 1.67 us 010b 4.17 us 110b 1.25 us 011b 3.13 us 111b 1.00 us 1. Voltage Ramp Time = maximum time to change VDDCR_CPU or VDDCR_NB by 12.5 mV rounded to the next higher encoding. For example, if the VDDCR_CPU regulator slew rate is 8 mV/us and the VDDCR_NB regulator slew rate is 5.5 mV/us, it takes the VDDCR_CPU regulator 1.56 us to change 12.5 mV and it takes the VDDCR_NB regulator 2.27 us to change 12.5 mV. In this case, BIOS should set this field to 2.5 us.
3:0	Reserved.

D18F3xDC Clock Power/Timing Control 2

Bits	Description
31	**CnbCifClockGateEn: NBCIF clock gating enable**. Read-write. Reset: 0. BIOS: 1. Specifies whether dynamic clock gating on the NBCIF is enabled. 1=Enabled. 0=Disabled. See 2.5.4.2 [NB Clock Ramping].
30	**NbClockGateEn: northbridge clock gating enable**. Read-write. Reset: 0. BIOS: 0. Specifies whether dynamic clock gating on the NB is enabled. 1=Enabled. 0=Disabled. See 2.5.4.2 [NB Clock Ramping].
29:27	**NbClockGateHyst: northbridge clock gating hysteresis**. Read-write. Reset: 0. Reset: 0. BIOS: 011b. Specifies how long hardware waits after the IFQ is empty and all cores are in a non-C0 state before gating the NB clocks. Bits / Time / Bits / Time 000b 0 100b 3 us 001b 320 ns 101b 5 us 010b 500 ns 110b 10 us 011b 1 us 111b 20 us See 2.5.4.2 [NB Clock Ramping].

26:20	**NbPs0NclkDiv: NCLK divisor**. Read-write. Reset: Product-specific. BIOS: See 2.5.4.1.1. Specifies the NCLK divisor when in NBP0.
	
	• The clock divisor can be calculated using the following table:
	Bits — Resulting Clock Divisor
	07h-00h — Reserved
	3Fh-08h — NbPs0NclkDiv * 0.25
	5Fh-40h — ((NbPs0NclkDiv - 40h) * 0.5) + 16
	7Fh-60h — NbPs0NclkDiv - 40h
	• 50% clock duty cycles are obtained at integer and half-integer divisors only. For example, /2, /2.5, /3.0, and /3.5 give 50% clock duty cycles whereas /3.25 does not.
	• Divisor examples:
	  • Example: If D18F3xDC[NbPs0NclkDiv] = 0Ch = 12d, then the NCLK divisor = 12 * 0.25 = /3.0
	• The NCLK COF can be calculated using the following equation:
	  • COF = (main PLL frequency specified by D18F3xD4[MainPllOpFreqId]) / clock divisor.
	  • Example: If D18F3xD4[MainPllOpFreqId] = 10h = 3.2 GHz and D18F3xDC[NbPs0NclkDiv] = 3Eh = /15.5, then the NCLK COF = 206.45 MHz.
	• Writes that change the value of this field cause NCLK to transition to the new divisor if the processor is currently in NBP0. This occurs regardless of the state of D18F6x90[NbPsCap].
	• Software may only change the value of this field if either:
	  • The sequence described in 2.9.3 [DCT/DRAM Initialization and Resume] has not been run, or
	  • DRAM has been placed into self-refresh. See D18F2x90[EnterSelfRef].
	
	See 2.5.4.1 [NB P-states].

19	**NclkFreqDone: NCLK frequency change done**. Read-only. Reset: 0. 1=NCLK frequency change complete. 0=NCLK frequency change in progress.
18:12	**NbPs0Vid**: **NB VID**. Read-write. Reset: Product-specific. BIOS: See 2.5.4.1.1. Specifies the VID for VDDCR_NB when in NBP0. Writes to this field cause the VID being output for VDDCR_NB to change if the processor is currently in NBP0. This occurs regardless of the state of D18F6x90[NbPsCap]. See the AMD Voltage Regulator Specification, #40182 for encodings. See 2.5.4.1 [NB P-states]. Writing this field while D18F6x90[NbPsCtrlDis] == 0 may result in undefined behavior. Whenever this field is written, software must wait the RampTime specified by D18F3xD8[VSRampSlamTime] before clearing D18F6x90[NbPsCtrlDis] to 0, changing the value of D18F3xDC[NbPs0NclkDiv], or changing the value of D18F6x90[NbPsForceSel].
11	Reserved.
10:8	**PstateMaxVal: P-state maximum value**. Read-write. IF ((D18F3xE8[HtcCapable] == 1) && (D18F3x64[HtcTmpLmt] != 0) && (D18F3x64[HtcPstateLimit] > PstateMaxVal)) THEN BIOS: D18F3x64[HtcPstateLimit]. ENDIF. Reset: specified by the reset state of MSRC001_00[6B:64][PstateEn]; the reset value is the highest P-state number corresponding to the MSR in which PstateEn is set. For example, if MSRC001_0064 and MSRC001_0065 have this bit set and the others do not, the reset value of PstateMaxVal is 1; if PstateEn is not set in any of these MSRs, the reset value of PstateMaxVal is 0. This field specifies the highest P-state value (lowest performance state) supported by the hardware. See MSRC001_0061[PstateMaxVal].
7:0	Reserved.

D18F3xE4 Thermtrip Status

Bits	Description
31	**SwThermtp: software THERMTRIP**. RAZ; write-1-only; cleared-by-hardware. Reset: 0. Writing a 1 to this bit position induces a THERMTRIP event. This bit returns 0 when read. This is a diagnostic bit, and it should be used for testing purposes only.
30:6	Reserved.
5	**ThermtpEn: THERMTRIP enable**. Read-only. Reset: Product-specific. 1=The THERMTRIP state as specified in section 2.10.3.2 [THERMTRIP] is supported by the processor.
4	Reserved.
3	**ThermtpSense: THERMTRIP sense**. Read-only. Cold reset: 0. 1=The processor temperature exceeded the THERMTRIP value (regardless as to whether the THERMTRIP state (ThermtpEn) is enabled). This bit is also set when the diagnostic bit SwThermtp = 1.
2	Reserved.
1	**Thermtp: THERMTRIP**. Read-only. Cold reset: 0. 1=The processor has entered the THERMTRIP state.
0	Reserved.

D18F3xE8 Northbridge Capabilities

Read-only. Unless otherwise specified, 1=The feature is supported by the processor; 0=The feature is not supported.

Bits	Description
31:13	Reserved.
12	**CmpCap: CMP capable**. Reset: Product-specific. Specifies the number of cores enabled on the device. Bits: 0b = 1; 1b = 2
11	Reserved.
10	**HtcCapable: HTC capable**. Reset: Product-specific.
9	**SvmCapable: SVM capable**. Reset: Product-specific.
8	**MctCap: memory controller (on the processor) capable**. Reset: 1.
7:5	**DdrMaxRate**. Specifies the maximum DRAM data rate that the processor is designed to support. Reset: Product-specific. 000b: No limit; 001b: Reserved; 010b: Reserved; 011b: 1066 MT/s; 100b: 800 MT/s; 101b: 667 MT/s; 110b: 533 MT/s; 111b: 400 MT/s
4:0	Reserved.

D18F3xF0 DEV Capability Header

See 2.8.3 [DMA Exclusion Vectors (DEV)]. DMA Exclusion Vectors (DEV) are contiguous arrays of bits in physical memory. There is no support for MMIO DEV tables. Each bit in the DEV table represents a 4KB page of physical memory; the DEV applies to accesses that target system memory and MMIO. The DEV table is packed as follows: bit[0] of byte 0 (pointed to by the DEV table base address, D18F3xF8_x0 and D18F3xF8_x1) controls the first 4K bytes of physical memory (starting at address 00_0000_0000h); bit[1] of byte 0 controls the second 4K bytes of physical memory; etc. When a DEV table bit is set to one, accesses to that physical page by external DMA devices is not allowed. If an external device attempts to access a protected physical page, then the processor master aborts the request.

In addition, the processor supports multiple protection domains. There is a DEV table for each protection domain. Link-defined UnitIDs or RequesterIDs may be assigned to the DEV of a specific protection domain through D18F3xF8_x2. DEV table walks for each protection domain are cached in the NB to reduce the number DEV table access to system memory.

The DEV function is configured through D18F3xF0, D18F3xF4, D18F3xF8, and an array of registers called D18F3xF8_DF[7:0], which are defined following D18F3xF8. D18F3xF4 [DEV Function/Index] and D18F3xF8 [DEV Data Port] are used to access F3xF8_DF[7:0]. The register number (i.e., the number that follows "_DF" in the register mnemonic) is specified by D18F3xF4[DevFunction]. In addition, D18F3xF8_x0, D18F3xF8_x1, and D18F3xF8_x2 are each instantiated multiple times, indexed by D18F3xF4[DevIndex]. Access to these registers is accomplished as follows:

- Reads:
  - Write the register number to D18F3xF4[DevFunction, DevIndex].
  - Read the register contents from D18F3xF8.

- Writes:
  - Write the register number to D18F3xF4[DevFunction, DevIndex].
  - Write the register contents to D18F3xF8.

IF (D18F3xE8[SvmCapable] == 0) THEN

Bits	Description
31:0	Reserved.

ELSE

Bits	Description
31:22	Reserved.
21	**IntCap: interrupt reporting capability**. Read-only. Reset: 0. 0=Interrupt reporting of DEV protection violations is not present on this device.
20	**MceCap: MCE reporting capability**. Read-only. Reset: 1. Indicates that machine check architecture reporting of DEV protection violations is present on this device.
19	Reserved.
18:16	**CapType: DEV capability block type**. Read-only. Reset: 000b. Specifies the layout of the Capability Block.
15:8	**CapPtr: capability pointer**. Read-only. Reset: 00h. Indicates that this is the last capability block.

7:0	**CapId: capability ID**. Read-only. Reset: 0Fh. Indicates a DEV capability block.

ENDIF.

D18F3xF4 DEV Function/Index

Reset: 0000_0000h.
IF (D18F3xE8[SvmCapable] == 0) THEN

Bits	Description
31:0	Reserved.

ELSE

Bits	Description
31:16	Reserved.
15:8	**DevFunction**. Read-write. See D18F3xF0. Valid values for this field are 00h through 07h. Writing invalid values may result in undefined behavior.
7:0	**DevIndex**. Read-write. See D18F3xF0. Valid values for this field are (1) 00h through (D18F3xF8_x3[NDomains] - 1) when either D18F3xF8_x0 or D18F3xF8_x1 are being accessed and (2) 00h through (D18F3xF8_x3[NMaps] + D18F3xF8_x3[NSrcMaps] - 1) when D18F3xF8_x2 is being accessed; this field is reserved for accesses to all other DEV configuration registers. Writing invalid values may result in undefined behavior.

ENDIF.

D18F3xF8 DEV Data Port

Reset: 0000_0000h. See D18F3xF0 for details about this port.

IF (D18F3xE8[SvmCapable] == 0) THEN

Bits	Description
31:0	Reserved.

ELSE

Bits	Description
31:0	**DevData**. Read-write. See D18F3xF8_x0, D18F3xF8_x1, D18F3xF8_x2, D18F3xF8_x3, D18F3xF8_x4, D18F3xF8_x5, D18F3xF8_x6, and D18F3xF8_x7.

ENDIF.

D18F3xF8_x0 DEV Base Address/Limit Low

Reset: 0000_0000h.

This register is instantiated multiple times, specified by D18F3xF8_x3[NDomains]. Each instantiation corresponds to a protection domain number, identical to D18F3xF4[DevIndex], which is the index to the instantiation. See D18F3xF0.

Bits	Description
31:12	**BaseAddress[31:12]: DEV table base address bits[31:12]**. Read-write. These bits are combined with D18F3xF8_x1[BaseAddress[39:32]] to specify the base address of the DEV table. The DEV table is required to be in either non-cacheable or write-through memory. Placing DEV tables in MMIO space is not supported. If any part of the DEV table is in other than system memory, then undefined behavior results.
11:7	Reserved.
6:2	**Size: DEV table size**. Read-write. These bits specify the size of the memory region that the DEV table covers. The corresponding DEV table size is 128KB*(2^Size). Bits Definition 08h-00h <4*2^Size> GB 1Fh-09h Reserved
1	**Protect: protect out-of-range addresses**. Read-write. 0=DMA accesses to addresses that are outside the range covered by the DEV table are allowed. 1=DMA accesses to addresses that are outside the range covered by the DEV table are protected.
0	**Valid: DEV table valid**. Read-write. 1=The DEV table for the protection domain specified by D18F3xF4[DevIndex] is enabled. 0=The DEV table is not enabled; all IO accesses from devices assigned to the corresponding protection domain are allowed.

D18F3xF8_x1 DEV Base Address/Limit High

Reset: 0000_0000h.

This register is instantiated multiple times, specified by D18F3xF8_x3[NDomains]. Each instantiation corresponds to a protection domain number, identical to D18F3xF4[DevIndex], which is the index to the instantiation. See D18F3xF0.

Bits	Description
31:8	Reserved.
7:0	**BaseAddress[39:32]: DEV table base address bits[39:32]**. Read-write. See D18F3xF8_x0[BaseAddress].

D18F3xF8_x2 DEV Map

Reset: 0000_0000h.

The DEV Map register maps internal unit IDs (see Table 98) to DEV protection domains. This register is instantiated the number of times specified D18F3xF8_x3[NMaps].

Table 98: Internal unit ID mapping

Unit ID	Internal Device
01h	GPU
03h:02h	Reserved
04h	Root Port (Bus 0, Device 4)
05h	Root Port (Bus 0, Device 5)
06h	Root Port (Bus 0, Device 6)

Table 98: Internal unit ID mapping

Unit ID	Internal Device
07h	Root Port (Bus 0, Device 7)
08h	FCH
1Fh:09h	Reserved

If Valid[x] is set, then the address of DMA requests received by the processor from an internal device with a UnitID of Unit[x] are checked against the DEV table of protection domain number Dom[x] to determine if the transaction is allowed. A UnitID can only be assigned to one protection domain. If a UnitID is assigned to more than one protection domain the results are undefined.

If the request doesn't match on any map register then the address of the request is checked against the DEV table of protection domain 0 to determine if the transaction is allowed.

Bits	Description
31:29	Reserved.
28:26	**Dom1: protection domain 1**. Read-write. This is the protection domain number assigned to Unit1.
25:23	Reserved.
22:20	**Dom0: protection domain 0**. Read-write. This is the protection domain number assigned to Unit0.
19:12	**BusNu: bus number**. Read-write.
11	**Valid1: UnitID 1 valid**. Read-write. 1=Enable DEV checking for Unit1 and Dom1.
10:6	**Unit1: internal UnitID 1**. Read-write.
5	**Valid0: UnitID 0 valid**. Read-write. 1=Enable DEV checking for Unit0 and Dom0.
4:0	**Unit0: internal UnitID 0**. Read-write.

D18F3xF8_x3 DEV Capabilities

Bits	Description
31:24	Reserved.
23:16	**NMaps: number of map registers implemented**. Read-only. Reset: 04h. Specifies the number of instantiations of D18F3xF8_x2 of the UID format.
15:8	**NDomains: number of protection domains implemented**. Read-only. Reset: 08h. Specifies the number of protection domains and the number of instantiations of D18F3xF8_x0 and D18F3xF8_x1.
7:0	**Revision: DEV register-set revision number**. Read-only. Reset: 02h. Indicates support for D18F3xF8_x4[SecureGfxMode].

D18F3xF8_x4 DEV Control

Reset: 0000_0000h.

Bits	Description
31:9	Reserved.

8	**SecureGfxMode: secure graphics mode**. Read-write; set-by-hardware. This bit is set when an SKINIT instruction is executed.1=All access to memory except for accesses to the frame buffer from the GPU are checked by the DEV if the DEV is enabled.
7	Reserved.
6	**DevTblWalkPrbDis: DEV table walk probe disable**. Read-write. 1=Disable probing of CPU caches during DEV table walks. This bit may be set to improve DEV cache table walk performance when the DEV is in non-cacheable or write-through memory.
5	**SlDev: secure loader DEV protection enable**. Read; write-0-only; set-by-hardware. This bit is set when an SKINIT instruction is executed. 1=The memory region associated with the SKINIT instruction is protected from DMA access.
4	**DevInv: invalidate DEV cache**. Read; write-1-only; cleared-when-done. 1=Invalidate the DEV table-walk cache. This bit is cleared by hardware when invalidation is complete.
3	**MceEn: MCE reporting enable**. Read-write. 1=Enable reporting of DEV protection violations through a machine check exception.
2	**IoDis: upstream IO disable**. Read-write; set-by-hardware. This bit is set when an SKINIT instruction is executed. 1=Upstream IO-space accesses are regarded as DEV protection violations.
1	Reserved.
0	**DevEn: DEV enable**. Read-write. 1=Enables DMA exclusion vector protection.

D18F3xF8_x5 DEV Error Status

Cold reset: 0000_0000h.

This register logs DEV protection violations. Bits[7:0], [ErrTypeDest, ErrTypeSrc, ErrTypeAccType], together form the error type field. When a DEV protection violation occurs, then ErrVal is set, the error type is logged, and, if there is an address associated with the transaction, ErrAddrVal is set and the address is recorded in D18F3xF8_x6 and D18F3xF8_x7.

Bits	Description
31	**ErrVal: error valid**. Read-write; set-by-hardware. 1=A valid DEV protection violation has been logged in this register.
30	**ErrOver: error overflow**. Read-write; set-by-hardware. 1=A DEV protection violation was detected while ErrVal was set for a prior violation. DEV protection violations detected while ErrVal is set are not logged in this register.
29	**ErrAddrVal: error address valid**. Read-write; set-by-hardware. 1=The address saved in D18F3xF8_x6 and D18F3xF8_x7 is the address associated with the error.
28:24	Reserved.
23:16	**ModelSpecErr: model specific error**. Read-only.
15:8	Reserved.
7:5	**ErrCodeDest: error code destination**. Read-write; set-by-hardware. Specifies the destination of the transaction that resulted in the protection violation. Bits / Definition / Bits / Definition 000b Generic (or could not be determined) / 100b IO space 001b DRAM / 101b Configuration 010b MMIO / 110b Reserved 011b Reserved / 111b Reserved

4:2	**ErrCodeSrc: error code source**. Read-write; set-by-hardware. Specifies the source of the transaction that resulted in the protection violation. Bits Definition Bits Definition 000b Generic (or could not be determined) 010b IO device 001b CPU 111b-011b Reserved
1:0	**ErrCodeAccType: error code access type**. Read-write; set-by-hardware. Specifies the access type of the transaction that resulted in the protection violation. Bits Definition Bits Definition 00b Generic (or could not be determined) 10b Write 01b Read 11b Read-modify-write

D18F3xF8_x6 DEV Error Address Low

Cold reset: 0000_0000h.

Bits	Description
31:3	**ErrAddr: error address bits[31:3]**. Read-write; set-by-hardware. See D18F3xF8_x5.
2:0	Reserved.

D18F3xF8_x7 DEV Error Address High

Cold reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**ErrAddr: error address bits[39:32]**. Read-write; set-by-hardware. See D18F3xF8_x5.

D18F3xFC CPUID Family/Model/Stepping

These values are identical to the values read out through CPUID Fn8000_0001_EAX.

Bits	Description
31:28	Reserved.
27:20	**ExtendedFamily: extended family**. Read-only. Reset: 5h.
19:16	**ExtendedModel: extended model**. Read-only. Reset: 0.
15:12	Reserved.
11:8	**BaseFamily**. Read-only. Reset: Fh.
7:4	**BaseModel**. Read-only. Reset: Product-specific.
3:0	**Stepping**. Read-only. Reset: Product-specific.

D18F3x128 Clock Power/Timing Control 3

Bits	Description
31:16	Reserved

15	**NbPsiVidEn: NB PSI_L enable**. Read-write. Reset: 0. BIOS: See 2.5.1.4.1 [PSI_L Bit]. This bit specifies how the PSI_L bit for VDDCR_NB is controlled. 1=See D18F3x128[NbPsiVid]. 0=The PSI_L bit is always 1.
14:8	**NbPsiVid**. Read-write. Reset: 0. If D18F3x128[NbPsiVidEn]==1, this field specifies a VID code that determines the state of the PSI_L bit for the VDDCR_NB plane. Whenever the VID code generated by the processor for the VDDCR_NB plane is less than (voltage is greater than) NbPsiVid, the PSI_L bit is sent as a 1. Whenever the VID code generated by the processor for the VDDCR_NB plane is greater than or equal to (voltage less than or equal to) NbPsiVid, the PSI_L bit is sent as a 0. See 2.5.1.4.1 [PSI_L Bit].
7	Reserved.
6:0	**C6Vid**. Read-write. Reset: Product-specific. Specifies the VID driven in the package PC6 state. See 2.5.3.2.3 [C-state Actions]. This field must be programmed within the limits specified by MSRC001_0071[MaxVid, MinVid].

D18F3x15C DPM Voltage Control

Bits	Description
31	Reserved.
30:24	**SclkVidLevel3**. Read-write. Reset: Product-specific. See D18F3x15C[SclkVidLevel0].
23	Reserved.
22:16	**SclkVidLevel2**. Read-write. Reset: Product-specific. See D18F3x15C[SclkVidLevel0].
15	Reserved.
14:8	**SclkVidLevel1**. Read-write. Reset: Product-specific. See D18F3x15C[SclkVidLevel0].
7	Reserved.
6:0	**SclkVidLevel0**. Read-write. Reset: Product-specific. Specifies a voltage level used for various NB power management features. See the AMD Voltage Regulator Specification, #40182 for encodings. If the VID code specified is 00h, software should consider the VID code invalid.

D18F3x17C In-Flight Queue Extended Buffer Allocation

Read-write. BIOS: See 2.9.3.5.
See D18F3x7C for buffer allocation requirements.

Bits	Description
31:14	Reserved.
13:8	**HiPriNPBC: high priority channel non-posted buffer count**. Reset: 0.
7:6	Reserved.
5:0	**HiPriPBC: high priority channel posted buffer count**. Reset: 0h.

D18F3x180 Extended NB MCA Configuration

Reset: 0000_0000h. This register is an extension of D18F3x44 [MCA NB Configuration].

Bits	Description
31:22	Reserved.

21	**SyncFloodOnCpuLeakErr: sync flood on CPU leak error enable**. Read-write. BIOS: 1. 1=A sync flood is generated when one of the cores encounters an uncorrectable error which cannot be contained to the process on the core.
20:8	Reserved.
7	**SyncFloodOnTgtAbtErr**. Read-write. 1=Enable sync flood on generated or received responses that indicate target aborts.
6	Reserved.
5	**DisPciCfgCpuMstAbtRsp**. Read-write. BIOS: 1b. 1=Disable MCA error reporting for master abort responses to CPU-initiated configuration accesses. It is recommended that this bit be set in order to avoid MCA exceptions being generated from master aborts for PCI configuration accesses, which are common during device enumeration.
4	**MstAbtChgToNoErrs**. Read-write. 1=Signal no errors instead of master abort in link response packets to IO devices on detection of a master abort condition. When MstAbtChgToNoErrs and D18F3x44[IoMstAbortDis] are both set, MstAbtChgToNoErrs takes precedence.
3	**DatErrChgToTgtAbt**. Read-write. 1=Signal target abort instead of data error in link response packets to IO devices (for Gen1 link compatibility).
2	**WDTCntSel[3]: watchdog timer count select bit[3]**. Read-write. BIOS: 0. See D18F3x44[WDTCntSel].
1:0	Reserved.

D18F3x188 NB Extended Configuration

Bits	Description
31:28	**FeArbCpuWeightOverHiPrio: NB front-end round-robin arbiter CPU weight over high priority channel**. Read-write. Reset: 4h. BIOS: 1h. This value is the number of times (out of 16) that the arbiter favors the CPU when both a CPU and the High Priority channel are requesting. FeArbCpuWeightOverHiPrio must be >= 1. When two CPUs are enabled each CPU is granted half of this value on average.
27:24	**FeArbCpuWeightOverLoPrio: NB front-end round-robin arbiter CPU weight over low priority channel**. Read-write. Reset: Bh. BIOS: Bh. This value is the number of times (out of 16) that the arbiter favors the CPU when both a CPU and the Low Priority channel are requesting. FeArbCpuWeightOverLoPrio must be >= FeArbCpuWeightOverHiPrio. When two CPUs are enabled each CPU is granted half of this value on average.
23	**EnCpuSerRdBehindIoRd: enable CPU serialization of sized reads behind IO reads**. Read-write. Reset: 0. BIOS: 0. 1=From the same CPU, sized reads are serialized behind IO reads. 0=From the same CPU, sized reads are not serialized behind IO reads.
22	**EnCpuSerRdBehindNpIoWr: enable CPU serialization of sized reads behind non-posted IO writes**. Read-write. Reset: 0. BIOS: See 2.9.3.5. 1=From the same CPU, sized reads are serialized behind non-posted IO writes. 0=From the same CPU, sized reads are not serialized behind non-posted IO writes.
21	**EnCpuSerWrBehindIoRd: enable CPU serialization of sized writes behind IO reads**. Read-write. Reset: 0. BIOS: 0. 1=From the same CPU, sized writes are serialized behind IO reads. 0=From the same CPU, sized writes are not serialized behind IO reads.
20:0	Reserved.

D18F3x1CC IBS Control

Per-core. Reset: 0000_0000h. This register can also be read from MSRC001_103A. D18F3x1CC is programmed by BIOS. The operating system reads the LVT offset from MSRC001_103A.

Bits	Description
31:9	Reserved.
8	**LvtOffsetVal: local vector table offset valid**. Read-write. BIOS: 1. 1=The offset in LvtOffset is valid.
7:4	Reserved.
3:0	**LvtOffset: local vector table offset**. Read-write. BIOS: 0h. This specifies the address of the IBS LVT entry in the APIC registers as follows: Bits Definition 3h-0h LVT address = <500h + LvtOffset<<4> Fh-4h Reserved See APIC[530:500].

D18F3x1E4 SBI Control

See 2.10.2 [Sideband Temperature Sensor Interface (SB-TSI)].

Bits	Description
31	**SbiRegWrDn: SBI register write done**. Read-only. Reset: 1b. 1=Write to the SBI registers through D18F3x1EC has completed. 0=Write to the SBI registers in progress.
30:7	Reserved.
6:4	**SbiAddr: SMBus-based sideband interface address**. Read-write. Cold reset: specified by the SA[2:0] strap pins (value matches the pins until the de-assertion of RESET_L for a cold reset only; value is not changed by a warm reset); 000b in products that do not include SA[2:0] pins. Specifies bits[3:1] of the SMBus address of the processor SBI ports. SMBus address bits [3:1] = {~SA[2],SA[1:0]}.
3:2	Reserved.
1	**SbTsiDis: SMBus-based sideband interface thermal sensor disable**. Read-only. Reset: Product-specific. 1=The processor does not support SMBus-based SBI thermal sensor protocol.
0	Reserved.

D18F3x1E8 SBI Address

Reset: 0000_0000h.

The SB-TSI registers can be directly accessed by the processor using D18F3x1E8 and D18F3x1EC. Access to these registers is accomplished as follows:

- Reads:
  - Write the register number to D18F3x1E8[SbiRegAddr].
  - Read the register contents from D18F3x1EC.

- Writes:
  - Write the register number to D18F3x1E8[SbiRegAddr].
  - Write all 32 bits to the register data to D18F3x1EC.

Bits	Description
31:8	Reserved.
7:0	**SbiRegAddr: SBI SMBus register address**. Read-write. This field specifies the 8-bit address of the SB-TSI register to access.

D18F3x1EC SBI Data

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**SbiRegDat0: SBI SMBus register data**. Read-write. This field specifies the data to be read or written to the SBI register selected by D18F3x1E8.

D18F3x1F0 Product Information

Bits	Description
31:16	Reserved.
15:0	**BrandId**. Read-only. Reset: Product-specific. Brand identifier. This is identical to CPUID Fn8000_0001_EBX[BrandId].

3.11 Device 18h Function 4 Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space.

D18F4x00 Device/Vendor ID

Reset: 1704_1022h.

Bits	Description
31:16	**DeviceID: device ID**. Read-only.
15:0	**VendorID: vendor ID**. Read-only.

D18F4x04 Status/Command

Read-only. Reset: 0000_0000h.

Bits	Description
31:16	**Status**.
15:0	**Command**.

D18F4x08 Class Code/Revision ID

Reset: 0600_0000h.

Bits	Description
31:8	**ClassCode**. Read-only. Provides the host bridge class code as defined in the PCI specification.
7:0	**RevID: revision ID**. Read-only.

D18F4x0C Header Type

Bits	Description
31:0	**HeaderTypeReg**. Value: 0080_0000h. The header type field indicates that there are multiple functions present in this device.

D18F4x34 Capabilities Pointer

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**CapPtr: capabilities pointer**. Read-only.

D18F4x118 C-state Control 1

Reset: 0000_0000h. BIOS: See 2.5.3.2.9.

D18F4x118 and D18F4x11C consist of eight identical 8-bit registers, one for each C-state Action Field (CAF) associated with an IO address that is read to enter C-states. See 2.5.3.2 [C-states].

Bits	Description
31:27	Reserved.
26:24	**CstAct3: C-state action field 3**. Read-write. Specifies the actions attempted by the core when software reads from the IO address specified by MSRC001_0073[CstateAddr + 3]. See Table 99.
23:19	Reserved.
18:16	**CstAct2: C-state action field 2**. Read-write. Specifies the actions attempted by the core when software reads from the IO address specified by MSRC001_0073[CstateAddr + 2]. See Table 99.
15:11	Reserved.
10:8	**CstAct1: C-state action field 1**. Read-write. Specifies the actions attempted by the core when software reads from the IO address specified by MSRC001_0073[CstateAddr + 1]. See Table 99.
7:3	Reserved.
2:0	**CstAct0: C-state action field 0**. Read-write. Specifies the actions attempted by the core when software reads from the IO address specified by MSRC001_0073[CstateAddr]. See Table 99.

Table 99: C-state action field definition

Bits	Description
7:1	Reserved.
0	**C6Enable**. Specifies whether the core attempts to enter CC6 and the package attempts to enter PC6. 1=Attempt to enter CC6 and PC6. 0=Do not attempt to enter CC6 and PC6. See 2.5.3.2.3 [C-state Actions].

D18F4x11C C-state Control 2

Reset: 0000_0000h.

Bits	Description
31:27	Reserved.
26:24	**CstAct7: C-state action field 7**. Read-write. Specifies the actions attempted by the core when software reads from the IO address specified by MSRC001_0073[CstateAddr + 7]. See Table 99.
23:19	Reserved.
18:16	**CstAct6: C-state action field 6**. Read-write. Specifies the actions attempted by the core when software reads from the IO address specified by MSRC001_0073[CstateAddr + 6]. See Table 99.
15:11	Reserved.
10:8	**CstAct5: C-state action field 5**. Read-write. Specifies the actions attempted by the core when software reads from the IO address specified by MSRC001_0073[CstateAddr + 5]. See Table 99.
7:3	Reserved.
2:0	**CstAct4: C-state action field 4**. Read-write. Specifies the actions attempted by the core when software reads from the IO address specified by MSRC001_0073[CstateAddr + 4]. See Table 99.

D18F4x120 C-state Policy Control 1

Reset: 1000_0000h.

Bits	Description
31	**CstateMsgDis: C-state messaging disable**. Read-write. Specifies whether a message is sent to the FCH when a package C-state transition occurs. 0=Send message. 1=Do not send message. See 2.5.3.2.4.2 [FCH Messaging] for details.
30:29	Reserved.
28:27	**CoreOffWriPri: CC6 write/read priority**. Read-write. Specifies the priority given to DRAM writes and reads when saving and restoring data to enter or exit the CC6 state. Bits Definition Bits Definition 00b Medium. 10b High. 01b Low. 11b Variable. See D18F2x118 for more details about DRAM transaction priorities. See 2.5.3.2.3.2 [Core C6 (CC6) State].
26:25	Reserved.
24	**DeepCstAllowMsgEn: deep C-state allow message enable**. Read-write. Specifies whether the processor prevents or allows access to PC6 based on FCH messaging. 0=Access to PC6 does not rely on FCH messaging. 1=Access to PC6 relies on FCH messaging. See 2.5.3.2.4.2 [FCH Messaging].

23	Reserved.
22:20	**FchTO: FCH timeout**. Read-write. Specifies the time to wait for the response from the FCH after requesting access to a package C-state. Bits Definition Bits Definition 000b Reserved 011b 100 us 001b 40 us 100b 200 us 010b 80 us 111b-101b Reserved See D18F4x120[DeepCstTOPol] and 2.5.3.2.4.2 [FCH Messaging].
19	**DeepCstTOPol: deep C-state timeout policy**. Read-write. Specifies the action to take if a timeout occurs while waiting for a response from the FCH after requesting access to PC6. 0=Prevent access to PC6. 1=Allow access to PC6. See 2.5.3.2.4.2 [FCH Messaging].
18	Reserved.
17:16	**DeepCstDMATrackEn**. Read-write. Specifies the type of DMA activity to track when determining whether to allow access to PC6. See D18F4x120[CstDMATrackHyst] and 2.5.3.2.4.1 [DMA Tracking] for more details. Bits Definition 00b DMA tracking disabled. 01b DMA tracking enabled, track coherent traffic only. 10b DMA tracking enabled, track non-coherent traffic only. 11b DMA tracking enabled, track both coherent and non-coherent traffic.
15:13	**CstDMATrackHyst: deep C-state DMA tracking hysteresis**. Read-write. Specifies the hysteresis time for DMA activity tracking. Bits Definition Bits Definition 000b Reserved 100b 80 us. 001b 10 us. 101b 100 us. 010b 20 us. 11xb Reserved. 011b 40 us. See 2.5.3.2.4.1 [DMA Tracking] for more details.
12:0	Reserved.

D18F4x124 C-state Monitor Control 1

Reset: 0000_0000h.

This register controls the various system activity monitors that can be used to limit or allow entry into certain C-states. See 2.5.3.2.4 [C-state Request Monitors].

Bits	Description
31	**IntMonWakeEn: timer tick interrupt monitor wake enable**. Read-write. Specifies whether a Timer Tick message from the FCH with encoding 1111b will automatically wake all cores from a C-state. 0=Do not wake cores. 1=Wake cores. See 2.5.3.2.4.3 [Interrupt Monitors].

<table>
<tr><td>30:27</td><td>IntMonIntrvl: timer tick interrupt monitor interval. Read-only. Specifies the time between timer tick interrupts last reported by the FCH. See 2.5.3.2.4.3 [Interrupt Monitors].
<table>
<tr><th>Bits</th><th>Definition</th><th>Bits</th><th>Definition</th></tr>
<tr><td>0000b</td><td>No interval reported</td><td>1000b</td><td>7 ms <= time < 8 ms</td></tr>
<tr><td>0001b</td><td>0 ms <= time < 1 ms</td><td>1001b</td><td>8 ms <= time < 9 ms</td></tr>
<tr><td>0010b</td><td>1 ms <= time < 2 ms</td><td>1010b</td><td>9 ms <= time < 10 ms</td></tr>
<tr><td>0011b</td><td>2 ms <= time < 3 ms</td><td>1011b</td><td>10 ms <= time < 15 ms</td></tr>
<tr><td>0100b</td><td>3 ms <= time < 4 ms</td><td>1100b</td><td>>= 15 ms</td></tr>
<tr><td>0101b</td><td>4 ms <= time < 5 ms</td><td>1101b</td><td>Reserved</td></tr>
<tr><td>0110b</td><td>5 ms <= time < 6 ms</td><td>1110b</td><td>Reserved</td></tr>
<tr><td>0111b</td><td>6 ms <= time < 7 ms</td><td>1111b</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>26:23</td><td>IntMonPkgC6Lmt: timer tick interrupt monitor package C6 limit. Read-write. BIOS: 1010b. Specifies the threshold for disallowing access to the package C6 state. See 2.5.3.2.4.3 [Interrupt Monitors].
IF (D18F4x124[TimerTickIntvlScale] == 1) THEN
<table>
<tr><th>Bits</th><th>Threshold</th></tr>
<tr><td>0000b</td><td>PC6 entry allowed</td></tr>
<tr><td>1001b-0001b</td><td>IntMonPkgC6Lmt * 50 us</td></tr>
<tr><td>1110b-1010b</td><td>((5 * IntMonPkgC6Lmt) - 40) * 50 us</td></tr>
<tr><td>1111b</td><td>2000 us</td></tr>
</table>
ELSE
<table>
<tr><th>Bits</th><th>Threshold</th></tr>
<tr><td>0000b</td><td>PC6 entry allowed</td></tr>
<tr><td>1001b-0001b</td><td>IntMonPkgC6Lmt * 1 ms</td></tr>
<tr><td>1110b-1010b</td><td>((5 * IntMonPkgC6Lmt) - 40) * 1 ms</td></tr>
<tr><td>1111b</td><td>40 ms</td></tr>
</table>
ENDIF.</td></tr>
<tr><td>22</td><td>IntMonPkgC6En: timer tick interrupt monitor package C6 enable. Read-write. BIOS: 0. Specifies whether the timer tick interrupt monitor is enabled for the package C6 state. 0=Disabled. 1=Enabled.</td></tr>
<tr><td>21:18</td><td>IntMonCC6Lmt: timer tick interrupt monitor core C6 limit. Read-write. BIOS: 0100b. Specifies the threshold for disallowing access to the core C6 state. See: D18F4x124[IntMonPkgC6Lmt]. See 2.5.3.2.4.3 [Interrupt Monitors].</td></tr>
<tr><td>17</td><td>IntMonCC6En: timer tick interrupt monitor core C6 enable. Read-write. BIOS: 1. Specifies whether the timer tick interrupt monitor is enabled for the core C6 state. 0=Disabled. 1=Enabled.</td></tr>
<tr><td>16</td><td>TrackTimerTickInterEn: track timer tick interrupt enable. Read-write. BIOS: 1. Specifies the timer tick interrupt monitor mode. 0=Interval mode. 1=Duration mode. See 2.5.3.2.4.3 [Interrupt Monitors].</td></tr>
<tr><td>15</td><td>TimerTickIntvlScale: timer tick interval scale. Read-write. BIOS: 1. See D18F4x124[IntMonPkgC6Lmt].</td></tr>
<tr><td>14</td><td>MonitorEnableMode: monitor enable mode. Read-write. Specifies when the C0 residency counter for each core stops counting in relation to core C-state entry. 0=The C0Timers stop counting on any non-CC0 C-state entry. 1=The C0Timers stop counting only when the core enters the CC6 state. See 2.5.3.2.4.4 [Residency Monitors].</td></tr>
<tr><td>13:7</td><td>Reserved.</td></tr>
<tr><td>6:4</td><td>C0MonCC6Cntr: C0 monitor core C6 counter. Read-write. Specifies the threshold for the C0 residency counter. See 2.5.3.2.4.4 [Residency Monitors].</td></tr>
</table>

3:1	**C0MonCC6Lmt: C0 monitor core C6 limit**. Read-write. Specifies the time threshold for the last C0 residency before incrementing the C0 residency counter. See 2.5.3.2.4.4 [Residency Monitors]. Bits Residency Bits Residency 000b Reserved 100b <= 800 us 001b <= 100 us 101b <= 1 ms 010b <= 200 us 110b <= 2 ms 011b <= 400 us 111b > 2 ms
0	**C0MonCC6En: C0 monitor core C6 enable**. Read-write. Specifies whether the C0 residency monitor is enabled for the CC6 state. 0=Disabled. 1=Enabled. See 2.5.3.2.4.4 [Residency Monitors].

D18F4x128 C-state Monitor Control 2

Reset: 0000_0000h.

Bits	Description
31:6	Reserved.
5:1	**NonC0MonCC6Lmt: non-C0 monitor for core C6 limit**. Read-write. Specifies the non-C0 residency time required on a single non-C0 entry before transitions to CC6 are allowed. See 2.5.3.2.4.4.2 [Non-C0 residency Monitor] for additional details. Bits Residency 00h <= Reserved 1Fh-01h <= NonC0Timer0 * 50 us
0	**NonC0MonCC6En: non-C0 monitor for core C6 enable**. Read-write. Specifies whether the non-C0 residency monitor is enabled for the CC6 state. 1=Enabled. 0=Disabled. See 2.5.3.2.4.4.2 [Non-C0 residency Monitor].

D18F4x12C C6 Base

Reset: 0000_0000h.

Bits	Description
31:12	Reserved.
11:0	**C6Base[35:24]**. Write-once. BIOS: See 2.9.6. Specifies the DRAM base address of the memory region used to store data for the CC6 C-state. See 2.5.3.2.3.2 [Core C6 (CC6) State] and 2.9.6 [DRAM CC6/PC6 Storage].

D18F4x134 C-state Monitor Control 3

Reset: 0000_0000h. See 2.5.3.2.4.3 [Interrupt Monitors].

Bits	Description
31:27	**IntRateCC6DecrRate: interrupt rate monitor CC6 decrement rate**. Read-write. IF (LOW_LATENCY) THEN BIOS: 18h. ELSE BIOS: 08h. ENDIF. Specifies the rate at which the CC6 interrupt counter is decremented. See D18F4x134[IntRatePkgC6DecrRate] for encodings.

26:24	**IntRateCC6BurstLen: interrupt rate monitor CC6 burst length**. Read-write. BIOS: 5. If the processor receives multiple interrupts within the time window specified by this field, the CC6 interrupt counter is only incremented by one. See D18F4x134[IntRatePkgC6BurstLen] for encodings.
23:20	**IntRateCC6Threshold: interrupt rate monitor CC6 threshold**. Read-write. BIOS: 4. Specifies the threshold the CC6 interrupt counter must reach before access to CC6 is denied.
19:16	**IntRateCC6MaxDepth: interrupt rate monitor CC6 maximum counter depth**. Read-write. BIOS: 5. Specifies the saturation point of the CC6 interrupt counter.
15:11	**IntRatePkgC6DecrRate: interrupt rate monitor PC6 decrement rate**. Read-write. BIOS: 0Ah. Specifies the rate at which the PC6 interrupt counter is decremented. Bits — Decrement Rate 00h — Reserved 1Fh-01h — <IntRatePkgC6DecrRate * 50> us
10:8	**IntRatePkgC6BurstLen: interrupt rate monitor PC6 burst length**. Read-write. BIOS: 1. If the processor receives multiple interrupts within the time window specified by this field, the PC6 interrupt counter is only incremented by one. Bits — Burst Length 000b — 80 ns 001b — 10 us 010b — 20 us 011b — 50 us 100b — 75 us 101b — 100 us 110b — 125 us 111b — 150 us
7:4	**IntRatePkgC6Threshold: interrupt rate monitor PC6 threshold**. Read-write. BIOS: 0. Specifies the threshold the PC6 interrupt counter must reach before access to PC6 is denied.
3:0	**IntRatePkgC6MaxDepth: interrupt rate monitor PC6 maximum counter depth**. Read-write. BIOS: 0. Specifies the saturation point of the PC6 interrupt counter.

D18F4x138 SMAF Code DID 0

Reset: 0000_0000h.

Bits	Description
31:29	Reserved.
28:24	**Smaf3Did**. Read-write. Specifies the clock divisor used when entering a low power state associated with SMAF3. See D18F4x13C[Smaf0Did].
23:21	Reserved.
20:16	**Smaf2Did**. Read-write. Specifies the clock divisor used when entering a low power state associated with SMAF2. See D18F4x13C[Smaf0Did].
15:13	Reserved.
12:8	**Smaf1Did**. Read-write. Specifies the clock divisor used when entering a low power state associated with SMAF1. See D18F4x13C[Smaf0Did].

7:5	Reserved.
4:0	**Smaf0Did**. Read-write. Specifies the clock divisor used when entering a low power state associated with SMAF0. See D18F3x80 and D18F3x84. Bits Divisor Bits Divisor 0Ch-00h Reserved 0Fh Clocks off 0Dh /512 1Fh-10h Reserved 0Eh Reserved See D18F4x1A8[SingleHaltCpuDid].

D18F4x13C SMAF Code DID 1

Reset: 0000_0000h.

Bits	Description
31:29	Reserved.
28:24	**Smaf7Did**. Read-write. Specifies the clock divisor used when entering a low power state associated with SMAF7. See D18F4x138[Smaf0Did].
23:21	Reserved.
20:16	**Smaf6Did**. Read-write. BIOS: 0Fh. Specifies the clock divisor used when entering a low power state associated with SMAF6. See D18F4x138[Smaf0Did].
15:13	Reserved.
12:8	**Smaf5Did**. Read-write. Specifies the clock divisor used when entering a low power state associated with SMAF5. See D18F4x138[Smaf0Did].
7:5	Reserved.
4:0	**Smaf4Did**. Read-write. BIOS: 0Fh. Specifies the clock divisor used when entering a low power state associated with SMAF4. See D18F4x138[Smaf0Did].

D18F4x164 Fixed Errata

Value: Product-specific.

Bits	Description
31:0	**FixedErrata**. See the *Revision Guide for AMD Family 14h Models 00h-0Fh Processors* for the definition of this field.

D18F4x1A4 C-state Monitor Mask

Reset: 0000_0000h.

Each bit in each field in this register corresponds to one C-state action field (see D18F4x118 and D18F4x11C). Bit 0 of each field corresponds to D18F4x118[CstAct0], bit 1 corresponds to D18F4x118[CstAct1] and so on. A monitor is masked for a CAF by setting the corresponding bit to 1. For example, to mask the timer tick monitor for D18F4x11C[CstAct4], set bit 12 in this register to a 1. See 2.5.3.2.4.5 [C-state Monitor Masking] for

additional details.

Bits	Description
31:24	**C0MonMask: C0 residency monitor mask**. Read-write. BIOS: FFh. Specifies whether the C0 residency monitor is masked. 1=Masked. 0=Unmasked. See 2.5.3.2.4.4 [Residency Monitors].
23:16	**NonC0MonMask: non-C0 residency monitor mask**. Read-write. BIOS: FFh. Specifies whether the non-C0 residency monitor is masked. 1=Masked. 0=Unmasked. See 2.5.3.2.4.4 [Residency Monitors].
15:8	**TimerTickMonMask: timer tick monitor mask**. Read-write. BIOS: FFh. Specifies whether the timer tick monitor is masked. 1=Masked. 0=Unmasked. See 2.5.3.2.4.3 [Interrupt Monitors].
7:0	**IntRateMonMask: interrupt rate monitor mask**. Read-write. BIOS: FCh. Specifies whether the interrupt rate monitor is masked. 1=Masked. 0=Unmasked. See 2.5.3.2.4.3 [Interrupt Monitors].

D18F4x1A8 CPU State Power Management Dynamic Control 0

Reset: 0000_0000h.

Bits	Description
31:30	Reserved.
29	**DramSrHystEnable: DRAM self-refresh hysteresis enable**. Read-write. BIOS: 1. Specifies whether the DRAM self-refresh hysteresis timer is enabled. 1=Enabled. 0=Disabled. See 2.5.5.1 [DRAM Self-Refresh].
28:26	**DramSrHyst: DRAM self-refresh hysteresis time**. Read-write. BIOS: 101b. Specifies the hysteresis time before DRAM is placed into self-refresh. See 2.5.5.1 [DRAM Self-Refresh]. Bits / Delay / Bits / Delay 000b Reserved 100b 5 us 001b Reserved 101b 10 us 010b Reserved 110b 20 us 011b 3 us 111b Reserved
25	**MemTriStateEn: memory clock tri-state enable**. Read-write. BIOS: 1. Specifies whether MEMCLK is tri-stated while DRAM is in self-refresh. 1=Tri-state MEMCLK. 0=Do not tri-state MEMCLK. See 2.5.5.1 [DRAM Self-Refresh] for details.
24	**DramSrEn: DRAM self-refresh enable**. Read-write. BIOS: See 2.9.3.5. 1=DRAM can be opportunistically placed into self-refresh. 0=DRAM is not placed into self-refresh. This bit should not be set until all cores have been enabled. See D18F0x68[Cpu1En] and 2.5.5.1 [DRAM Self-Refresh].
23	**PServiceTmrEn**. Read-write. BIOS: 1. 0=The PService timer is disabled. 1=The PService timer is enabled. See 2.5.3.2.7 [C-state initiated P-state Changes].
22:20	**PServiceTmr**. Read-write. BIOS: 001b. Specifies the expiration time of the PService timer. See 2.5.3.2.7 [C-state initiated P-state Changes]. Bits / Delay / Bits / Delay 000b Reserved 100b 800 us 001b 100 us 101b 1 ms 010b 200 us 110b 2 ms 011b 400 us 111b 5 ms

19:17	**PService: service P-state**. Read-write. BIOS: See 2.5.3.2.9. Specifies the PService state. See 2.5.3.2.7 [C-state initiated P-state Changes]. This field is an index into MSRC001_00[6B:64]. If this field is programmed to 0, the PService state corresponds to the P-state specified by MSRC001_0064. If this field is programmed to 1, the PService state corresponds to the P-state specified by MSRC001_0065, and so on. PService must be programmed to lowest-performance P-state displayed to the operating system or to any lower-performance P-state.
16	Reserved.
15	**CpuProbEn: CPU probe enable**. Read-write. BIOS: 0. Specifies the core frequency used to service probes when ramping from CC1 or PC1. See 2.5.3.2.5 [C-states and Probe Requests].
14:10	Reserved.
9:5	**AllHaltCpuDid**. Read-write. BIOS: 1Fh. Specifies the divisor used when entering PC1 with or without auto-Pmin. See 2.5.3.2.3.3 [Package C1 (PC1) State]. Bits Divisor 1Ch-00h Reserved. 1Dh 128 1Eh 512 1Fh Clocks off. See D18F4x1A8[SingleHaltCpuDid].
4:0	**SingleHaltCpuDid**. Read-write. BIOS: 1Eh. On a processor with multiple cores, this specifies the divisor used when ramping core clocks down after a single core has entered the clocks ramped state. Bits Divisor 1Ch-00h Reserved. 1Dh 128 1Eh 512 1Fh Clocks off. • If MSRC001_00[6B:64][CpuDidMSD] of the current P-state is greater than or equal to SingleHaltCpuDid, then no frequency change is made when entering the low-power state associated with this register. • The COF = (the frequency specified by D18F3xD4[MainPllOpFreqId]) / (the divisor specified by this field).

D18F4x1AC CPU State Power Management Dynamic Control 1

Bits	Description
31	Reserved.
30	**CstPminEn: C-state Pmin enable**. Read-write. Reset: 0. BIOS: See 2.5.3.2.9. Specifies whether auto-Pmin is enabled. 1=Enabled. 0=Disabled. See 2.5.3.2.7 [C-state initiated P-state Changes].
29	**CoreC6Dis: core C6 disable**. Read-write. Reset: 0. See D18F4x1AC[CoreC6Cap].
28	**PkgC6Dis: package C6 disable**. Read-write. Reset: 0. See D18F4x1AC[PkgC6Cap].

27	**CoreC6Cap: core C6 capable**. Read-only. Reset: Product-specific. Along with D18F4x1AC[CoreC6Dis], this field specifies whether the processor is capable of removing power from a core when in the CC6 state. If either D18F4x1AC[CoreC6Dis]==1 or D18F4x1AC[CoreC6Cap]==0, the processor cannot remove power from cores. If both D18F4x1AC[CoreC6Dis]==0 and D18F4x1AC[CoreC6Cap]==1, the processor can remove power from cores. See 2.5.3.2.3 [C-state Actions].
26	**PkgC6Cap: package C6 capable**. Read-only. Reset: Product-specific. Along with D18F4x1AC[PkgC6Dis], this field specifies whether the processor is capable of entering the PC6 state. If either D18F4x1AC[PkgC6Dis]==1 or D18F4x1AC[PkgC6Cap]==0, the processor cannot enter the PC6 state. If both D18F4x1AC[PkgC6Dis]==0 and D18F4x1AC[PkgC6Cap]==1, the processor can enter the PC6 state. See 2.5.3.2.3 [C-state Actions].
25:19	Reserved.
18:16	**PstateIdCoreOffExit**. Read-write. Reset: 0. BIOS: See 2.5.3.2.9. When exiting the package C6 state, the core transitions to the P-state specified by this register. See 2.5.3.2.7.2 [Exiting PC6]. If PC6 is enabled (see 2.5.3.2.9 [BIOS Requirements for C-state Initialization]), PstateIdCoreOffExit must be programmed to the lowest-performance P-state displayed to the operating system or to any lower-performance P-state. This P-state must have a core clock frequency of at least 400 MHz. Programming this field to 0 causes the core to transition to the last P-state requested by software when exiting package C6.
15:10	Reserved.
9:5	**C6Did: CC6 divisor**. Read-write. Reset: 0. BIOS: 1Fh. Specifies the divisor applied to the core when ramping clocks down for CC6. See 2.5.3.2.3 [C-state Actions] for details. Bits — Divisor 1Ch-00h — Reserved. 1Dh — 128 1Eh — 512 1Fh — Clocks off. See D18F4x1A8[SingleHaltDid].
4:0	Reserved.

3.12 Device 18h Function 5 Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space.

D18F5x00 Device/Vendor ID

Reset: 1718_1022h.

Bits	Description
31:16	**DeviceID: device ID**. Read-only.
15:0	**VendorID: vendor ID**. Read-only.

D18F5x04 Status/Command

Read-only. Reset: 0000_0000h.

Bits	Description
31:16	**Status**.
15:0	**Command**.

D18F5x08 Class Code/Revision ID

Reset: 0600_0000h.

Bits	Description
31:8	**ClassCode**. Read-only. Provides the host bridge class code as defined in the PCI specification.
7:0	**RevID: revision ID**. Read-only.

D18F5x0C Header Type

Bits	Description
31:0	**HeaderTypeReg**. Value: 0080_0000h. The header type field indicates that there are multiple functions present in this device.

D18F5x34 Capabilities Pointer

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**CapPtr: capabilities pointer**. Read-only.

3.13 Device 18h Function 6 Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space.

D18F6x00 Device/Vendor ID

Reset: 1716_1022h.

Bits	Description
31:16	**DeviceID: device ID**. Read-only.
15:0	**VendorID: vendor ID**. Read-only.

D18F6x04 Status/Command

Read-only. Reset: 0000_0000h.

Bits	Description
31:16	**Status**.
15:0	**Command**.

D18F6x08 Class Code/Revision ID

Reset: 0600_0000h.

Bits	Description
31:8	**ClassCode**. Read-only. Provides the host bridge class code as defined in the PCI specification.
7:0	**RevID: revision ID**. Read-only.

D18F6x0C Header Type

Bits	Description
31:0	**HeaderTypeReg**. Value: 0080_0000h. The header type field indicates that there are multiple functions present in this device.

D18F6x34 Capabilities Pointer

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**CapPtr: capabilities pointer**. Read-only.

D18F6x50 Configuration Register Access Control

Reset: 0000_0006h.

Bits	Description
31:2	Reserved.
1	**CfgAccAddrMode: configuration access address mode**. Read-write. BIOS: 0. Specifies the register range accessible by the SMU. 1=Access to only bus 0 device 18h, function 6, offsets 54h to 7Fh and 154h to 17Fh is supported. 0=Access to all bus 0 device 18h offsets is supported.
0	Reserved.

D18F6x54 DRAM Arbitration Control FEQ Collision

Reset: 0000_0000h.

Bits	Description
31	**PpMode: Protection Period Mode**. Read-write. BIOS: 0. 1=The protection period for this register is based on counting 32-byte data packets of display requests. 0=The protection period for this register is based on counting NCLK cycles.
30:24	Reserved.
23:16	**FeqHiPrio: FEQ high priority**. Read-write. BIOS: 08h. Protection period since the display opened a rank/bank before a pending FEQ high priority request to the same rank/bank pair can be considered eligible for arbitration.
15:8	**FeqMedPrio: FEQ medium priority**. Read-write. BIOS: 10h. Protection period since the display opened a rank/bank before a pending FEQ medium priority request to the same rank/bank pair can be considered eligible for arbitration.
7:0	**FeqLoPrio: FEQ low priority**. Read-write. BIOS: 20h. Protection period since the display opened a rank/bank before a pending FEQ low priority request to the same rank/bank pair can be considered eligible for arbitration.

D18F6x58 DRAM Arbitration Control Display Collision

Reset: 0000_0000h.

Bits	Description
31:24	**DispUrgPrio: display urgent priority**. Read-write. BIOS: 00h. Number of NCLK cycles since FEQ opened a rank/bank before a pending display urgent priority request to the same rank/bank pair can be considered eligible for arbitration.
23:16	**DispHiPrio: display high priority**. Read-write. BIOS: 10h. Number of NCLK cycles since FEQ opened a rank/bank before a pending display high priority request to the same rank/bank pair can be considered eligible for arbitration.
15:8	**DispMedPrio: display medium priority**. Read-write. BIOS: 20h. Number of NCLK cycles since FEQ opened a rank/bank before a pending display medium priority request to the same rank/bank pair can be considered eligible for arbitration.
7:0	**DispLoPrio: display low priority**. Read-write. BIOS: 40h. Number of NCLK cycles since FEQ opened a rank/bank before a pending display low priority request to the same rank/bank pair can be considered eligible for arbitration.

D18F6x5C DRAM Arbitration Control FEQ Write Protect

Reset: 0000_0000h.

Bits	Description
31	**PpMode: Protection Period Mode**. Read-write. BIOS: 0b. 1=The protection period for this register is based on counting 32-byte data packets of display requests. When set, D18F6x6C[FeqHiPrio, FeqMedPrio and FeqLoPrio] must not equal FFh. 0=The protection period for this register is based on counting NCLK cycles.
30:24	Reserved.

23:16	**FeqHiPrio: FEQ high priority**. Read-write. BIOS: 08h. Protection period since display write was arbitrated before a pending FEQ high priority read request is eligible for arbitration.
15:8	**FeqMedPrio: FEQ medium priority**. Read-write. BIOS: 10h. Protection period since display write was arbitrated before a pending FEQ medium priority read request is eligible for arbitration.
7:0	**FeqLoPrio: FEQ low priority**. Read-write. BIOS: 20h. Protection period since display write was arbitrated before a pending FEQ low priority read request is eligible for arbitration.

D18F6x60 DRAM Arbitration Control Display Write Protect

Reset: 0000_0000h.

Bits	Description
31:24	**DispUrgPrio: display urgent priority**. Read-write. BIOS: 00h. Number of NCLK cycles since FEQ write was arbitrated before a pending display urgent priority read request is eligible for arbitration.
23:16	**DispHiPrio: display high priority**. Read-write. BIOS: 08h. Number of NCLK cycles since FEQ write was arbitrated before a pending display high priority read request is eligible for arbitration.
15:8	**DispMedPrio: display medium priority**. Read-write. BIOS: 10h. Number of NCLK cycles since FEQ write was arbitrated before a pending display medium priority read request is eligible for arbitration.
7:0	**DispLoPri: display low priority**. Read-write. BIOS: 20h. Number of NCLK cycles since FEQ write was arbitrated before a pending display low priority read request is eligible for arbitration.

D18F6x64 DRAM Arbitration Control FEQ Read Protect

Reset: 0000_0000h.

Bits	Description
31	**PpMode: Protection Period Mode**. Read-write. BIOS: 0b. 1=The protection period for this register is based on counting 32-byte data packets of display requests. When set, D18F6x6C[FeqHiPrio, FeqMedPrio and FeqLoPrio] must not equal FFh. 0=The protection period for this register is based on counting NCLK cycles.
30:24	Reserved.
23:16	**FeqHiPrio: FEQ high priority**. Read-write. BIOS: 04h. Protection period since display read was arbitrated before a pending FEQ high priority write request is eligible for arbitration.
15:8	**FeqMedPrio: FEQ medium priority**. Read-write. BIOS: 08h. Protection period since display read was arbitrated before a pending FEQ medium priority write request is eligible for arbitration.
7:0	**FeqLoPrio: FEQ low priority**. Read-write. BIOS: 10h. Protection period since display read was arbitrated before a pending FEQ low priority write request is eligible for arbitration.

D18F6x68 DRAM Arbitration Control Display Read Protect

Reset: 0000_0000h.

Bits	Description
31:24	**DispUrgPrio: display urgent priority**. Read-write. BIOS: 00h. Number of NCLK cycles since FEQ read was arbitrated before a pending display urgent priority write request is eligible for arbitration.

23:16	**DispHiPrio: display high priority**. Read-write. BIOS: 04h. Number of NCLK cycles since FEQ read was arbitrated before a pending display high priority write request is eligible for arbitration.
15:8	**DispMedPrio: display medium priority**. Read-write. BIOS: 08h. Number of NCLK cycles since FEQ read was arbitrated before a pending display medium priority write request is eligible for arbitration.
7:0	**DispLoPrio: display low priority**. Read-write. BIOS: 10h. Number of NCLK cycles since FEQ read was arbitrated before a pending display low priority write request is eligible for arbitration.

D18F6x6C DRAM Arbitration Control FEQ Fairness Timer

Reset: 00FF_FFFFh.

Bits	Description
31:24	Reserved.
23:16	**FeqHiPrio: FEQ high priority**. Read-write. BIOS: 20h. This field defines the number of NCLK cycles a high priority FEQ request must wait before its priority gets elevated to be arbitrated immediately.
15:8	**FeqMedPrio: FEQ medium priority**. Read-write. BIOS: 40h. This field defines the number of NCLK cycles a medium priority FEQ request must wait before its priority gets elevated to be arbitrated immediately.
7:0	**FeqLoPrio: FEQ low priority**. Read-write. BIOS: 80h. This field defines the number of NCLK cycles a low priority FEQ request must wait before its priority gets elevated to be arbitrated immediately.

D18F6x70 DRAM Arbitration Control Display Fairness Timer

Reset: FFFF_FFFFh.

Bits	Description
31:24	**DispUrPrio: display urgent priority**. Read-write. BIOS: 00h. This field defines the number of NCLK cycles an urgent priority display request must wait before its priority gets elevated to be arbitrated immediately.
23:16	**DispHiPrio: display high priority**. Read-write. BIOS: 20h. This field defines the number of NCLK cycles a high priority display request must wait before its priority gets elevated to be arbitrated immediately.
15:8	**DispMedPrio: display medium priority**. Read-write. BIOS: 40h. This field defines the number of NCLK cycles a medium priority display request must wait before its priority gets elevated to be arbitrated immediately.
7:0	**DispLoPrio: display low priority**. Read-write. BIOS: 80h. This field defines the number of NCLK cycles a low priority display request must wait before its priority gets elevated to be arbitrated immediately.

D18F6x74 Dram Idle Page Close Limit

Reset: 0000_001Eh.

Bits	Description
31:5	Reserved.
4:0	**IdleLimit: idle limit**. Read-write. BIOS: 1Eh. If D18F2x90[DynPageCloseEn]=0 then this field multiplied by 4 defines the number of NB clock cycles a page will be kept open after last page hit. Bits: Definition 1Eh-00h: <4*IdleLimit> NCLK delay. 1Fh: Reserved.

D18F6x78 Dram Prioritization and Arbitration Control

Reset: 0000_0000h.

Bits	Description
31:16	Reserved.
15:8	**DbeCmdThrottle: Dram controller back-end command throttle**. Read-write. BIOS: See 2.9.3.5. This field defines a limit for 64-byte read or write requests pending in the DRAM controller back-end. If the number of pending requests reaches or exceeds this limit, the DRAM controller front-end applies command throttling. Bits: Limit 00h: Throttling is disabled. FFh-01h: <DbeCmdThrottle> request(s)
7:6	Reserved.
5:4	**GlcEosDet: display end of stream detection**. Read-write. BIOS: 11b. Specifies the number of NCLK cycles that the display queue must be empty before a display end of stream event is detected. Upon a display end of stream event the read/write protection counter is reset if it was started by a display request. Bits: Limit 00b: <2^GlcEosDet> NCLK cycle(s)
3	**DispArbCtrl: display arbitration control**. Read-write. BIOS: 0. 1=Display requests, which are delayed by a bank-state-machine-full condition, block lower-priority front-end queue requests from being arbitrated. 0=Display requests, which are delayed by a bank-state-machine-full condition, do not block lower-priority front-end queue requests from being arbitrated.
2	**FeqDbePrioEn: FEQ DBE priority enable**. Read-write. BIOS: 1. 1=The DRAM controller back-end priority condition is asserted for high priority front-end queue requests. 0=The DRAM controller back-end priority condition is never asserted for front-end queue requests.
1:0	**DispDbePrioEn: display DBE priority enable**. Read-write. BIOS: 11b. Specifies which type of display requests assert the priority condition in the DRAM controller back-end. Bits: Definition 00b: No display request 01b: Display refresh request of any display priority 10b: Any display request of urgent priority 11b: Display refresh request of urgent priority

D18F6x90 NB P-state Config Low

See 2.5.4.1 [NB P-states].

Bits	Description
31	**NbPsCap: NB P-state capable**. Read-only. Reset: Product-specific. 1=The processor is capable of making NB P-state transitions. 0=The processor is not capable of making NB P-state transitions.
30	**NbPsCtrlDis: NB P-state control disable**. IF (D18F6x90[NbPsLock]) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. Specifies whether hardware is responsible for requesting NB P-state transitions. 0=Hardware dynamically requests NB P-state transitions during run-time without additional input from software. 1=Hardware does not make NB P-state transition requests. Whenever this bit is set to 1, hardware automatically makes a transition to NBP0.
29	**NbPsForceSel: NB P-state force selection**. Read-write. Reset: 0. Specifies the target NB P-state for any forced NB P-state transition. 1=Target NBP1. 0=Target NBP0. See D18F6x90[NbPsForceReq].
28	**NbPsForceReq: NB P-state force request**. Read-write. Reset: 0. If D18F6x98[NbPsTransInFlight]==0 when this bit is set, a forced NB P-state transition to the state specified by D18F6x90[NbPsForceSel] is initiated. This transition occurs regardless of the state of D18F6x90[NbPsCtrlDis]. If D18F6x98[NbPsTransInFlight]==1 when this bit is set to 1, the NB P-state transition request is queued until the transition currently in flight finishes.
27:21	Reserved.
20	**NbPsLock: NB P-state lock**. Read; write-1-only. Reset: Product-specific. This field controls the writability of several other fields relating to NB P-states. See the following fields for details: • D18F6x90[NbPsCtrlDis]. • D18F6x90[NbPs1Vid]. • D18F6x90[NbPs1NclkDiv].
19:17	Reserved.
16	**NbPs1GnbSlowIgn: NB P-state ignore GPU slow signal**. Read-write. Reset: 0. IF (GpuEnabled) THEN BIOS: 0. ELSE BIOS: 1. ENDIF. 0=The GPU driver specifies a level of GPU activity that can cause an NB P-state transition. 1=GPU activity is not taken into account when determining whether to make an NB P-state transition.
15	Reserved.
14:8	**NbPs1Vid**. IF (D18F6x90[NbPsLock]) THEN Read-only. ELSE Read-write. ENDIF. Reset: Product-specific. BIOS: See 2.5.4.1.1. Specifies the VID code output by the processor for VDDCR_NB when in NBP1. Writes to this field take effect on the next transition from NBP0 to NBP1. See the AMD Voltage Regulator Specification, #40182 for encodings. Writing this field while D18F6x90[NbPsCtrlDis] == 0 may result in undefined behavior. Whenever this field is written, software must wait the RampTime specified by D18F3xD8[VSRampSlamTime] before clearing D18F6x90[NbPsCtrlDis] to 0, changing the value of D18F6x90[NbPs1NclkDiv], or changing the value of D18F6x90[NbPsForceSel].

7	Reserved.

6:0	**NbPs1NclkDiv: NBP1 NCLK divisor**. IF (D18F6x90[NbPsLock]) THEN Read-only. ELSE Read-write. ENDIF. Reset: Product-specific. BIOS: See 2.5.4.1.1. Specifies the divisor applied to NCLK when in NBP1.
	
	• Writes that change the value of this field cause NCLK to transition to the new divisor if the processor is currently in NBP1.
	• Software may only change the value of this field if either:
	  • The sequence described in 2.9.3 [DCT/DRAM Initialization and Resume] has not been run, or
	  • DRAM has been placed into self-refresh. See D18F2x90[EnterSelfRef].
	
	See: D18F3xDC[NbPs0NclkDiv].

D18F6x94 NB P-state Config High

Reset: 0000_0000h. See 2.5.4.1 [NB P-states].

Bits	Description
31:29	**NbPs0ResTmrMin: NBP0 minimum residency timer**. Read-write. Upon transitioning from NBP1 to NBP0, transitions back to NBP1 are blocked until the amount of time specified by this register has passed. Bits — Residency 000b — 0 001b — 120 ns 010b — 100 us 011b — 500 us 100b — 1 ms 101b — 5 ms 110b — 10 ms 111b — 50 ms
28:26	**NbPs1ResTmrMin: NBP1 minimum residency timer**. Read-write. Upon transitioning from NBP0 to NBP1, transitions back to NBP0 are blocked until the amount of time specified by this register has passed. See: D18F6x94[NbPs0ResTmrMin].
25:23	**NbPsC0Timer: NB P-state C0 timer**. Read-write. BIOS: 100b. Specifies the time any core must be in C0 before a transition from NBP1 to NBP0 is triggered. Bits — Residency 000b — 0 001b — 120 ns 010b — 25 us 011b — 50 us 100b — 100 us 101b — 200 us 110b — 500 us 111b — 1 ms
22:20	**NbPsNonC0Timer: NB P-state non-C0 timer**. Read-write. Specifies the time all cores must be in a non-C0 C-state before a transition from NBP0 to NBP1 is triggered. See: D18F6x94[NbPsC0Timer].
19:5	Reserved.

4	**NbPsNoTransOnDma: NB P-state no transitions on DMA**. Read-write. 1=DMA traffic prevents NB P-state transitions. 0=DMA traffic do not affect NB P-state transitions.
3	**CpuPstateThrEn: CPU P-state threshold enable**. Read-write. BIOS: 1. Specifies whether core P-states are used as a threshold for NB P-state transitions. 1=Core P-states are used as a threshold. 0=Core P-states are not used.
2:0	**CpuPstateThr: CPU P-state threshold**. Read-write. BIOS: 1. When D18F6x94[CpuPstateThrEn]==1, this field specifies the core P-state number that acts as a threshold for NB P-states.

D18F6x98 NB P-state Control and Status

Reset: 0000_0000h. See 2.5.4.1 [NB P-states].

Bits	Description
31	**NbPsDbgEn: NB P-state debug enable**. Read-write. For any registers that change context based on NB P-states, this field specifies what causes a context swap. 0=Register context is selected by the current NB P-state. 1=Register context is selected by D18F6x98[NbPsCsrAccSel] regardless of the current NB P-state. See 2.9.3.4.7 [NB P-states for DCT/DRAM Initialization and Training].
30	**NbPsCsrAccSel: NB P-state register accessibility select**. Read-write. If D18F6x98[NbPsDbgEn]==1, this field specifies the context of any registers that have context swaps based on NB P-state. 0=Registers access the NBP0 context. 1=Registers access the NBP1 context. See 2.9.3.4.7 [NB P-states for DCT/DRAM Initialization and Training].
29:3	Reserved.
2	**NbPs1Act: NB P-state 1 active**. Read-only; updated-by-hardware. Specifies the current NB P-state. 0=NB is currently in NBP0. 1=NB is currently in NBP1.
1	**NbPs1ActSts: NB P-state 1 active status**. Read; set-by-hardware; write-1-to-clear. Specifies whether the NB has ever transitioned to NBP1 since the last time this field was cleared by software. 1=NB transitioned to NBP1.
0	**NbPsTransInFlight: NB P-state transition in flight**. Read-only. Specifies whether an NB P-state transition is in process. 1=NB P-state transition is occurring. 0=All NB P-state transitions completed.

D18F6x9C NCLK Reduction Control

Reset: 0000_00FFh. See 2.5.4.2 [NB Clock Ramping].

Bits	Description
31:9	Reserved.
8	**NclkRampWithDllRelock**. Read-write. BIOS: 1. Specifies whether NCLK ramps up in parallel or serially with the DDR PHY DLL being relocked when exiting NB clock ramping. 0=The DDR PHY DLL is relocked after NCLK is ramped up. 1=The DDR PHY DLL relock is started at the same NCLK begins ramping up. This bit can only be programmed to 1 if all of the following are true, or undefined behavior results: • D18F2x90[DisDllShutdownSR]==0. • (the main PLL frequency specified by D18F3xD4[MainPllOpFreqId]) / (the divisor specified by D18F6x9C[NclkRedDiv]) >= 100 MHz.

7	**NclkRedSelfRefrAlways: NCLK reduction during self-refresh always**. Read-write. BIOS: 1. Specifies which C-state the package must be in to allow NCLK to ramp down when DRAM enters self-refresh. 0=PC6. 1=PC1 or PC6. See 2.5.4.2 [NB Clock Ramping] for details.
6:0	**NclkRedDiv: NCLK reduction divisor**. Read-write. BIOS: 60h. Specifies the divisor used for NCLK when NCLK is ramped down while DRAM is in self-refresh. The following divisors may be created: Bits — Divisor 1Fh-00h — Reserved 20h — /8 3Fh-21h — Reserved 40h — /16 5Fh-41h — Reserved 60h — /32 7Fh-61h — Reserved NclkRedDiv must be programmed to a divisor greater than (lower frequency than) D18F3xDC[NbPs0NclkDiv] and D18F6x90[NbPs1NclkDiv] or undefined behavior may result.

D18F6x[B8:B0] Package C-state Residency

Reset: 0000_0000h. Each counter in D18F6x[B8:B0] applies to one package C-state and is enabled as follows:

Register	Function	Enable bit
D18F6xB0	Any non-C0 package state	D18F6xE0[PkgNonC0ResEn]
D18F6xB4	Package C1 with auto-Pmin	D18F6xE0[PkgC1ResEn]
D18F6xB8	Package C6	D18F6xE0[PkgC6ResEn]

Bits	Description
31:0	**PkgCstateResidency**. Read-write; updated-by-hardware. Specifies the time the package is in the corresponding state. Hardware increments this field once for every 80 ns spent in the package C-state while the specified enable bit is set to 1. Any write by software clears this register to 0. See 2.5.3.2.3 [C-state Actions].

D18F6xE0 Power Management Residency Counter Enable

Reset: 0000_0000h.

Bits	Description
31:3	Reserved.
2	**PkgC6ResEn: package C6 residency counter enable**. Read-write. See D18F6x[B8:B0].
1	**PkgC1ResEn: package C1 with auto-Pmin residency counter enable**. Read-write. See D18F6x[B8:B0].
0	**PkgNonC0ResEn: package non-C0 residency counter enable**. Read-write. See D18F6x[B8:B0].

D18F6x1[2C:10] NB FIFO Offset Configuration 7:0

Reset: 0000_0000h.

Bits	Description
31	**NclkFreqType: NCLK frequency type**. Read-write. 1=NclkFreq specifies a clock divider. 0=Reserved.
30:24	**NclkFreq: NCLK frequency**. Read-write. Specifies the NCLK frequency divider when NclkFifoOffset is applied.
23	**LclkFreqType: LCLK frequency type**. Read-write. 1=LclkFreq specifies a clock divider. 0=Reserved.
22:16	**LclkFreq: LCLK frequency**. Read-write. Specifies the LCLK frequency divider when LclkFifoOffset is applied.
15	**Enable: offset enable**. Read-write. 1=Offset enabled.
14:7	Reserved.
6:4	**LclkFifoOff: LCLK FIFO offset**. Read-write. Specifies the LCLK FIFO offset to apply.
3	Reserved.
2:0	**NclkFifoOff: NCLK FIFO offset**. Read-write. Specifies the NCLK FIFO offset to apply.

3.14 Device 18h Function 7 Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space.

D18F7x00 Device/Vendor ID

Reset: 1719_1022h.

Bits	Description
31:16	**DeviceID: device ID**. Read-only.
15:0	**VendorID: vendor ID**. Read-only.

D18F7x04 Status/Command

Read-only. Reset: 0000_0000h.

Bits	Description
31:16	**Status**.
15:0	**Command**.

D18F7x08 Class Code/Revision ID

Reset: 0600_0000h.

Bits	Description
31:8	**ClassCode**. Read-only. Provides the host bridge class code as defined in the PCI specification.
7:0	**RevID: revision ID**. Read-only.

D18F7x0C Header Type

Bits	Description
31:0	**HeaderTypeReg**. Value: 0080_0000h.The header type field indicates that there are multiple functions present in this device.

D18F7x34 Capabilities Pointer

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**CapPtr: capabilities pointer**. Read-only.

3.15 Internal System Management Unit (SMU) Registers

See section 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention.

To read SMU registers, software performs the following sequence:

1. If reading a 16-bit register, write the address to D0F0x64_x4D[SmuAddr]. If reading a 32-bit register, write the address + 1 to D0F0x64_x4D[SmuAddr]. This must be done for all reads.
2. Clear D0F0x64_x4D[ReqType] to 0.
3. If D0F0x64_x4D[ReqToggle]==0, set it to 1. If D0F0x64_x4D[ReqToggle]==1, clear it to 0.
4. Read data from D0F0x64_x4E[SmuReadData].

To write data to a 16-bit SMU register, software performs the following sequence:

1. Write the address to D0F0x64_x4D[SmuAddr].
2. Write the data to D0F0x64_x4D[WriteData].
3. Set D0F0x64_x4D[ReqType] to 1.
4. If D0F0x64_x4D[ReqToggle]==0, set it to 1. If D0F0x64_x4D[ReqToggle]==1, clear it to 0.

To write data to a 32-bit SMU register, software performs the following sequence:

1. Perform steps 1-4 of the 16-bit write process above using the lower 16 bits of data.
2. Program D0F0x64_x4D[SmuAddr]+=1.
3. Write the upper 16 bits of data to D0F0x64_x4D[WriteData].
4. If D0F0x64_x4D[ReqToggle]==0, set it to 1. If D0F0x64_x4D[ReqToggle]==1, clear it to 0.

In each sequence above, the writes to D0F0x64_x4D may be combined into a single write.

SMUx03 MCU IRQ

Reset: 0000_0000h.

Bits	Description
31:11	Reserved.

10:3	**ServiceIndex**. Read-write. Specifies the service index used by microcontroller firmware when interrupted by software. See 2.12.1.1 [Software Interrupts].
2	**IntDone: interrupt done**. Read-only. Specifies whether the interrupt requested by writing SMUx03[IntReq] has completed. 0=Interrupt service in progress. 1=Interrupt service complete. This field is cleared by hardware when software requests an interrupt by writing to SMUx03[IntReq]. See 2.12.1.1 [Software Interrupts].
1	**IntAck: interrupt acknowledge**. Read-only. Specifies whether the interrupt has been acknowledged by the microcontroller. 0=Not acknowledged. 1=Acknowledged. This field is cleared by hardware when software requests an interrupt by writing to SMUx03[IntReq]. See 2.12.1.1 [Software Interrupts].
0	**IntReq: interrupt request**. Read-write. When software sets this field to 1, an interrupt is triggered to the microcontroller. No additional software interrupts can be triggered until SMUx03[IntDone]==1. Software must clear this field to 0 before setting it to 1. See 2.12.1.1 [Software Interrupts].

SMUx05 SMU Data RAM

Reset: 0000_0000h.

Bits	Description
31:0	**McuRam**. Read-write. This field is used to access SMU RAM. Reading or writing this field causes SMUx0B[MemAddr] to increment by 4.

SMUx0B SMU RAM Address

Reset: 0000_0000h. The index/data pair registers SMUx0B and SMUx05 are used to access the registers SMUx0B_x[8FFF:8000]. To read or write to one of these registers, the index is written first into the index register SMUx0B and then the data is read or written by reading or writing the data register SMUx05. These registers definitions represent SMU RAM locations defined for a specific purpose. GMMx100 and GMMx104 may also be used to access SMUx0B_x[8FFF:8000] by subtracting 8000h from the index before programming GMMx100.

Bits	Description
15:0	**MemAddr**. Read-write. Specifies the SMU RAM address accessed.

SMUx0B_x8600 DMA Transaction Array 1

Reset: xxxx_xxxxh.

Bits	Description
31:24	**TransactionCount**. Read-write.
23:16	**MemAddr[15:8]**. Read-write.
15:8	**MemAddr[7:0]**. Read-write.
7:0	**Txn1MBusAddr[7:0]**. Read-write.

SMUx0B_x8604 DMA Transaction Array 2

Reset: xxxx_xxxxh.

Bits	Description
31:24	**Txn1MBusAddr[15:8]**. Read-write.
23:16	**Txn1MBusAddr[23:16]**. Read-write.
15:8	**Txn1MBusAddr[31:24]**. Read-write.
7:0	**Txn1TransferLength[7:0]**. Read-write.

SMUx0B_x8608 DMA Transaction Array 3

Reset: xxxx_xxxxh.

Bits	Description
31:30	**Txn1Tsize**. Read-write.
29:24	**Txn1TransferLength[13:8]**. Read-write.
23:20	**Txn1Spare**. Read-write.
19	**Txn1Overlap**. Read-write.
18	**Txn1Static**. Read-write.
17:16	**Txn1Mode**. Read-write.
15:8	**Txn2Mbusaddr70**. Read-write.
7:0	**Txn2Mbusaddr158**. Read-write.

SMUx0B_x860C DMA Transaction Array 4

Reset: xxxx_xxxxh.

Bits	Description
31:24	**Txn2MBusAddr2316**. Read-write.
23:16	**Txn2MBusAddr3124**. Read-write.
15:8	**Txn2TransferLength70**. Read-write.
7:6	**Txn2Tsize**. Read-write.
5:0	**Txn2TransferLength138**. Read-write.

SMUx0B_x8610 DMA Transaction Array 5

Reset: xxxx_xxxxh.

Bits	Description
31:28	**Txn2Spare**. Read-write.
27	**Txn2Overlap**. Read-write.
26	**Txn2Static**. Read-write.
25:24	**Txn2Mode**. Read-write.
23:16	**Txn3MBusAddr70**. Read-write.
15:8	**Txn3MBusAddr158**. Read-write.
7:0	**Txn3MBusAddr2316**. Read-write.

SMUx0B_x8614 DMA Transaction Array 6

Reset: xxxx_xxxxh.

Bits	Description
31:24	**Txn3MBusAddr3124**. Read-write.
23:16	**Txn3TransferLength70**. Read-write.
15:14	**Txn3Tsize**. Read-write.
13:8	**Txn3TransferLength138**. Read-write.
7:4	**Txn3Spare**. Read-write.
3	**Txn3Overlap**. Read-write.
2	**Txn3Static**. Read-write.
1:0	**Txn3Mode**. Read-write.

SMUx0B_x8618 DMA Transaction Array 7

Reset: xxxx_xxxxh.

Bits	Description
31:24	**Txn4MBusAddr70**. Read-write.
23:16	**Txn4MBusAddr158**. Read-write.
15:8	**Txn4MBusAddr2316**. Read-write.
7:0	**Txn4MBusAddr3124**. Read-write.

SMUx0B_x861C DMA Transaction Array 8

Reset: xxxx_xxxxh.

Bits	Description
31:24	**Txn4TransferLength70**. Read-write.
23:22	**Txn4Tsize**. Read-write.
21:16	**Txn4TransferLength138**. Read-write.
15:12	**Txn4Spare**. Read-write.
11	**Txn4Overlap**. Read-write.
10	**Txn4Static**. Read-write.
9:8	**Txn4Mode**. Read-write.
7:0	**Txn5Mbusaddr70**. Read-write.

SMUx0B_x8620 DMA Transaction Array 9

Reset: xxxx_xxxxh.

Bits	Description
31:24	**Txn5MBusAddr158**. Read-write.

23:16	**Txn5MBusAddr2316**. Read-write.
15:8	**Txn5MBusAddr3124**. Read-write.
7:0	**Txn5TransferLength70**. Read-write.

SMUx0B_x8624 DMA Transaction Array 10

Reset: xxxx_xxxxh.

Bits	Description
31:30	**Txn5Tsize**. Read-write.
29:24	**Txn5TransferLength138**. Read-write.
23:20	**Txn5Spare**. Read-write.
19	**Txn5Overlap**. Read-write.
18	**Txn5Static**. Read-write.
17:16	**Txn5Mode**. Read-write.
15:8	**Txn6MBusAddr70**. Read-write.
7:0	**Txn6MBusAddr158**. Read-write.

SMUx0B_x8628 DMA Transaction Array 11

Reset: xxxx_xxxxh.

Bits	Description
31:24	**Txn6MBusAddr2316**. Read-write.
23:16	**Txn6MBusAddr3124**. Read-write.
15:8	**Txn6TransferLength70**. Read-write.
7:6	**Txn6Tsize**. Read-write.
5:0	**Txn6TransferLength138**. Read-write.

SMUx0B_x862C DMA Transaction Array 12

Reset: xxxx_xxxxh.

Bits	Description
31:28	**Txn6Spare**. Read-write.
27	**Txn6Overlap**. Read-write.
26	**Txn6Static**. Read-write.
25:24	**Txn6Mode**. Read-write.
23:16	**Txn7MBusAddr70**. Read-write.
15:8	**Txn7MBusAddr158**. Read-write.
7:0	**Txn7MBusAddr2316**. Read-write.

SMUx0B_x8630 DMA Transaction Array 13

Reset: xxxx_xxxxh.

Bits	Description
31:24	**Txn7MBusAddr3124**. Read-write.
23:16	**Txn7TransferLength70**. Read-write.
15:14	**Txn7Tsize**. Read-write.
13:8	**Txn7TransferLength138**. Read-write.
7:4	**Txn7Spare**. Read-write.
3	**Txn7Overlap**. Read-write.
2	**Txn7Static**. Read-write.
1:0	**Txn7Mode**. Read-write.

SMUx0B_x8634 DMA Transaction Array 14

Reset: xxxx_xxxxh.

Bits	Description
31:24	**Txn8MBusAddr70**. Read-write.
23:16	**Txn8MBusAddr158**. Read-write.
15:8	**Txn8MBusAddr2316**. Read-write.
7:0	**Txn8MBusAddr3124**. Read-write.

SMUx0B_x8638 DMA Transaction Array 15

Reset: xxxx_xxxxh.

Bits	Description
31:24	**Txn8TransferLength70**. Read-write.
23:22	**Txn8Tsize**. Read-write.
21:16	**Txn8TransferLength138**. Read-write.
15:12	**Txn8Spare**. Read-write.
11	**Txn8Overlap**. Read-write.
10	**Txn8Static**. Read-write.
9:8	**Txn8Mode**. Read-write.
7:0	**Txn9MBusAddr70**. Read-write.

SMUx0B_x863C DMA Transaction Array 16

Reset: xxxx_xxxxh.

Bits	Description
31:24	**Txn9MBusAddr158**. Read-write.

23:16	**Txn9MBuAaddr2316**. Read-write.
15:8	**Txn9MBusAddr3124**. Read-write.
7:0	**Txn9TransferLength70**. Read-write.

SMUx0B_x8640 DMA Transaction Array 17

Reset: xxxx_xxxxh.

Bits	Description
31:30	**Txn9Tsize**. Read-write.
29:24	**Txn9TransferLength138**. Read-write.
23:20	**Txn9Spare**. Read-write.
19	**Txn9Overlap**. Read-write.
18	**Txn9Static**. Read-write.
17:16	**Txn9Mode**. Read-write.
15:8	**Txn10MBusAddr70**. Read-write.
7:0	**Txn10MBusAddr158**. Read-write.

SMUx0B_x86[A0:50:step4] DMA Scratch Data 21-1

Reset: xxxx_xxxxh.

Bits	Description
31:0	**Data**. Read-write.

3.16 Fixed Configuration Space (FCR)

See section 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention.

To read FCR space, software performs the following sequence:

1. Program the following registers:
   - SMUx0B_x8600[TransactionCount] = 1.
   - SMUx0B_x8600[MemAddr[15:8]] = 86h.
   - SMUx0B_x8600[MemAddr[7:0]] = 50h.
   - SMUx0B_x8600[Txn1MBusAddr[7:0]] = bits [7:0] of the FCR register's address.
   - SMUx0B_x8604[Txn1MBusAddr[15:8]] = bits [15:8] of the FCR register's address.
   - SMUx0B_x8604[Txn1MBusAddr[23:16]] = bits [23:16] of the FCR register's address.
   - SMUx0B_x8604[Txn1MBusAddr[31:24]] = bits [31:24] of the FCR register's address.
   - SMUx0B_x8604[Txn1TransferLength[7:0]] = 4.
   - SMUx0B_x8608[Txn1Tsize] = 3.
   - SMUx0B_x8608[Txn1TransferLenth[13:8]] = 0.
   - SMUx0B_x8608[Txn1Overlap] = 0.
   - SMUx0B_x8608[Txn1Static] = 1.
   - SMUx0B_x8608[Txn1Mode] = 0.
2. If the FCR register's address begins with FE00h, interrupt the SMU with Service Index 0Dh. If the FCR register's address begins with FF30h, interrupt the SMU with Service Index 0Bh. Wait for the interrupt to complete. See 2.12.1.1 [Software Interrupts].

3. Read data from SMUx0B_x8650 in SMUx0B_x86[A0:50:step4].

To write FCR space, software performs the following sequence:

1. Write data to SMUx0B_x8650 (see SMUx0B_x86[A0:50:step4]).
2. Program the following registers:
   - SMUx0B_x8600[TransactionCount] = 1.
   - SMUx0B_x8600[MemAddr[15:8]] = 86h.
   - SMUx0B_x8600[MemAddr[7:0]] = 50h.
   - SMUx0B_x8600[Txn1MBusAddr[7:0]] = bits [7:0] of the FCR register's address.
   - SMUx0B_x8604[Txn1MBusAddr[15:8]] = bits [15:8] of the FCR register's address.
   - SMUx0B_x8604[Txn1MBusAddr[23:16]] = bits [23:16] of the FCR register's address.
   - SMUx0B_x8604[Txn1MBusAddr[31:24]] = bits [31:24] of the FCR register's address.
   - SMUx0B_x8604[Txn1TransferLength[7:0]] = 4.
   - SMUx0B_x8608[Txn1Tsize] = 3.
   - SMUx0B_x8608[Txn1TransferLenth[13:8]] = 0.
   - SMUx0B_x8608[Txn1Overlap] = 0.
   - SMUx0B_x8608[Txn1Static] = 1.
   - SMUx0B_x8608[Txn1Mode] = 1.
3. Interrupt the SMU with Service Index 0Bh and wait for the interrupt to complete as described in 2.12.1.1 [Software Interrupts].
4. Read data from SMUx0B_x8650 (see SMUx0B_x86[A0:50:step4]).

FCRxFE00_6000 NB P-state Configuration 0

Bits	Description
31:21	Reserved.
20:14	**NbPs1Vid**. Value: Product-specific. Specifies the initial NBP1 voltage. See 2.5.4.1.1 [BIOS Requirements for NB P-state Initialization During DRAM Training].
13:7	**NbPs0Vid**. Value: Product-specific. Specifies the initial NBP0 voltage. See 2.5.4.1.1 [BIOS Requirements for NB P-state Initialization During DRAM Training].
6:0	Reserved.

FCRxFE00_6002 NB P-state Configuration 1

Bits	Description
31:19	Reserved.
18:12	**NbPs1VidHigh**. Value: Product-specific. Specifies a VID code used when calculating NB P-state voltages. See 2.5.4.1.1 [BIOS Requirements for NB P-state Initialization During DRAM Training].
11:5	**NbPs1VidAddl: NBP1 VID additional**. Value: Product-specific. Specifies a VID code used when calculating NB P-state voltages. See 2.5.4.1.1 [BIOS Requirements for NB P-state Initialization During DRAM Training].
4:0	Reserved.

FCRxFE00_600E Clock Configuration

Bits	Description
31:6	Reserved.
5:0	**MainPllOpFreqIdStartup**. Value: Product-specific. Specifies the COF of the main PLL following a cold reset. See D18F3xD4[MainPllOpFreqId].

FCRxFE00_7006 NB P-state Configuration 2

Bits	Description
31:26	Reserved.
25:21	**MaxNbFreqAtMinVid: maximum at NB frequency at minimum NB VID**. Value: Product-specific. Specifies a frequency used when calculating NB P-state frequencies. See 2.5.4.1.1 [BIOS Requirements for NB P-state Initialization During DRAM Training]. Bits / Description: 00001b-00000b Reserved.; 00010b 100 MHz; 00101b-00011b Reserved.; 00110b 200 MHz; 01001b-00111b Reserved.; 01010b 267 MHz; 01101b-01011b Reserved.; 01110b 333 MHz; 11111b-01111b Reserved.
20:14	**NbPs1NclkDiv**. Value: Product-specific. Specifies the initial NBP1 frequency. See 2.5.4.1.1 [BIOS Requirements for NB P-state Initialization During DRAM Training].
13:0	Reserved.

FCRxFE00_7009 NB P-state Configuration 3

Bits	Description
31:9	Reserved.
8:2	**NbPs0NclkDiv**. Value: Product-specific. Specifies the initial NBP0 frequency. See 2.5.4.1.1 [BIOS Requirements for NB P-state Initialization During DRAM Training].
1:0	Reserved.

FCRxFE00_7070 GNB Configuration Miscellaneous 0

Bits	Description
31:7	Reserved.
6:3	**GnbIdleAdjustVid**. Value: Product-specific. Specifies the voltage offset that is subtracted from VDDCR_NB as specified by FCRxFF30_0191[GfxIdleVoltChgMode, GfxIdleVoltChgMode]. Voltage offset = GnbIdleAdjustVid * 12.5 mV.
2:0	Reserved.

FCRxFE00_70A2 Power Configuration Miscellaneous

Bits	Description
31:20	Reserved.
19:18	**PcieGen2Vid**. Value: Product-specific. Specifies the voltage required for generation 2 PCIe operation. See 2.5.1.3 [BIOS Requirements for Power Plane Initialization]. This field indexes into D18F3x15C as follows: Bits — VID code 00b — D18F3x15C[SclkVidLevel0] 01b — D18F3x15C[SclkVidLevel1] 10b — D18F3x15C[SclkVidLevel2] 11b — D18F3x15C[SclkVidLevel3]
17:0	Reserved.

FCRxFF30_0191 SCLK Power Management Control

Reset: 0000_F812h.

Bits	Description
31:18	Reserved.
17	**GfxIdleVoltChgMode: GFX idle voltage change mode**. Read-write. Specifies whether the GPU must be power gated or clock gated before hardware can reduce the VDDCR_NB voltage. 0=Power gated. 1=Clock gated. See 2.5.1.4.2 [Alternate Low Power Voltages].
16	**GfxIdleVoltChgEn: GFX idle voltage change enable**. Read-write. BIOS: 1. Specifies whether hardware reduces the VDDCR_NB voltage when the GPU is clock gated or power gated. 0=No voltage reduction. 1=Reduce voltage as specified by FCRxFF30_0191[GfxIdleVoltChgMode] and FCRxFE00_7070[GnbIdleAdjustVid]. See 2.5.1.4.2 [Alternate Low Power Voltages].
15:0	Reserved.

3.17 GPU Memory Mapped Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming conventions.

Some reserved registers in the GMM space have side effects when read. Software should not access any GMM register other than those listed.

3.17.1 Memory Mapped SMU Registers

GMMx100 RCU Indirect Index

Reset: 0000_0000h. The index/data pair registers GMMx100 and GMMx104 are used to access the registers GMMx104_x[FFF:000]. To read or write to one of these registers, the address is written first into the address register GMMx100 and then the data is read or written by reading or writing the data register GMMx104. The index/data pair registers SMUx0B and SMUx05 can also be used to access GMMx104_x[FFF:000] by adding 8000h to the index before programming SMUx0B. See SMUx0B_x[8FFF:8000] for the register definitions.

Bits	Description
31:0	**RcuIndIndex: RCU indirect index**. Read-write.

GMMx104 RCU Indirect Data

Reset: 0000_0000h. See GMMx100.

Bits	Description
31:0	**RcuIndData: RCU indirect data**. Read-write.

GMMx670 GPU Control

Bits	Description
31:2	Reserved.
1	**GpuDis: GPU disable**. Read-only. Reset: Product-specific. Specifies whether the GPU is disabled. 0=GPU is enabled. 1=GPU is disabled.
0	Reserved.

GMMx770 CG Voltage Control

Reset: 0000_0006h. See 2.5.1.5.2 [Software-Initiated Voltage Transitions].

Bits	Description
31:5	Reserved.
4	**VoltageForceEn**. Read-write. See D0F0x64_x6A[VoltageForceEn].
3	**VoltageChangeEn**. Read-write. See D0F0x64_x6A[VoltageChangeEn].
2:1	**VoltageLevel**. Read-write. See D0F0x64_x6A[VoltageLevel].
0	**VoltageChangeReq**. Read-write. See D0F0x64_x6A[VoltageChangeReq].

GMMx774 CG Voltage Status

Reset: 0000_0000h. See 2.5.1.5.2 [Software-Initiated Voltage Transitions].

Bits	Description
31:3	Reserved.
2:1	**CurrentVoltageLevel**. Read-only. See D0F0x64_x6B[CurrentVoltageLevel].
0	**VoltageChangeAck**. Read-only. See D0F0x64_x6B[VoltageChangeAck].

GMMx2024 GMC Frame Buffer Location

Reset: 0000_0000h. The register along with GMMx2898 [GMC Frame Buffer Offset] specifies the base address of the frame buffer in GPU physical memory and system memory.

Bits	Description
31:16	**Top: frame buffer region GPU limit address[39:24]**. Read-write. Specifies the GPU address of the top of the frame buffer region.
15:0	**Base: frame buffer GPU base address[30:24]**. Read-write. Specifies the GPU base address of the frame buffer.

GMMx2028 GMC VM Noncoherent System Memory Top

BIOS: 0000_0000h.

Bits	Description
31:18	Reserved.
17:0	**SysTop[39:22]**. Read-write. Reset: 0.

GMMx202C GMC VM Noncoherent System Memory Bottom

BIOS: 0003_FFFFh.

Bits	Description
31:18	Reserved.
17:0	**SysBot[39:22]**. Read-write. Reset: 0.

GMMx2[8D8,77C] GMC DRAM Timing

Reset: 1919_0A0Ah.

Bits	Description
31:24	**RasMActWr: Trc - Trcdw**. Read-write. BIOS: D18F2xF4_x40[Trc]-D18F2xF4_x40[Trcd]+6.
23:16	**RasMActRd: Trc - Trcdr**. Read-write. BIOS: D18F2xF4_x40[Trc]-D18F2xF4_x40[Trcd]+6.
15:8	**ActWr: Trcdw**. Read-write. BIOS: D18F2xF4_x40[Trcd] + 5.
7:0	**ActRd: Trcdr**. Read-write. BIOS: D18F2xF4_x40[Trcd] + 5.

GMMx2[8DC,780] GMC DRAM Timing 2

Reset: 0D14_0A23h.

Bits	Description
31:24	**BusTurn: bus turnaround time**. Read-write. BIOS: (D18F2x84[Tcwl] + D18F2xF4_x41[Twtr] + D18F2x8C[TrwtTO] + 15)/2.

23:16	**WrPlusRp: Twr + Trp**. Read-write.
	IF (D18F2x84[Twr] < 100b) THEN
	BIOS: D18F2xF4_x40[Trp] + D18F2x84[Twr] + 8.
	ELSE
	BIOS: D18F2xF4_x40[Trp] + 2*D18F2x84[Twr] + 4.
	ENDIF.

15:8	**Rp: Trp - 1**. Read-write. BIOS: D18F2xF4_x40[Trp] + 4.
7:0	**Ras2Ras: Trc -1**. Read-write. BIOS: D18F2xF4_x40[Trc] + 10.

GMMx28[34:1C:step8] GMC DCT CS Base Address

The information in these registers is copied from D18F2x[4C:40] [DRAM CS Base Address].

Table 100: GMC CS base address register mapping and reset values

Register	Function	Reset	DCT Register
GMMx281C	CS0	0000_0001h	D18F2x40
GMMx2824	CS1	0000_0081h	D18F2x44
GMMx282C	CS2	0000_0101h	D18F2x48
GMMx2834	CS3	0000_0181h	D18F2x4C

Bits	Description
31:28	Reserved.
27:19	**BaseAddr[35:27]: normalized physical base address bits [35:27]**. Read-write. BIOS: D18F2x[4C:40][BaseAddr[35:27]].
18:14	Reserved.
13:5	**BaseAddr[21:13]: normalized physical base address bits [21:13]**. Read-write. BIOS: D18F2x[4C:40][BaseAddr[21:13]].
4:1	Reserved.
0	**CSEnable: chip select enable**. Read-write. BIOS: D18F2x[4C:40][CsEnable].

GMMx28[40:3C] GMC DCT CS[3:0] Mask

Reset: 01F8_3CE0h. The information in these registers is copied from D18F2x[64:60] [DRAM CS Mask].

Table 101: GMC CS mask register mapping

Register	Function	DCT Register
GMMx283C	CS[1:0]	D18F2x60
GMMx2840	CS[3:2]	D18F2x64

Bits	Description
31:29	Reserved.
28	Reserved.
27:19	**AddrMask[35:27]: normalized physical address mask bits [35:27]**. Read-write. BIOS: D18F2x[64:60][AddrMask[35:27]].
18:14	Reserved.
13:5	**AddrMask[21:13]: normalized physical address mask bits [21:13]**. Read-write. BIOS: D18F2x[64:60][AddrMask[21:13]].
4:0	Reserved.

GMMx284C GMC DCT Bank Address Mapping

Reset: 0002_0077h.

Bits	Description
31:20	Reserved.
19	**BankSwap: bank swap**. Read-write. BIOS: D18F2xA8[BankSwap].
18:17	Reserved.
16	**BankSwizzleMode: bank swizzle mode**. Read-write. BIOS: D18F2x94[BankSwizzleMode].
15:8	Reserved.
7:4	**Dimm1AddrMap: DIMM 1 address map**. Read-write. BIOS: D18F2x80[Dimm1AddrMap].
3:0	**Dimm0AddrMap: DIMM 0 address map**. Read-write. BIOS: D18F2x80[Dimm0AddrMap].

GMMx2858 GMC DRAM Control 2

Reset: 0000_0000h.

Bits	Description
31:10	Reserved.
9	**DctSelBankSwap: select DRAM bank swap address**. Read-write. BIOS: D18F2x114[DctSelBankSwap].
8:0	Reserved.

GMMx285C GMC DRAM Hole Address

Reset: 0000_0000h.

Bits	Description
31:24	**DramHoleBase[31:24]: DRAM hole base address**. Read-write. BIOS: D18F1xF0[DramHoleBase].
23:16	Reserved.
15:7	**DramHoleOffset[31:23]: DRAM hole offset address**. Read-write. BIOS: D18F1xF0[DramHoleOffset].
6:1	Reserved.
0	**DramHoleValid**. Read-write. BIOS: D18F1xF0[DramHoleValid].

GMMx2864 DRAM Address Swizzle

BIOS: 3210_0876h. Swizzle address bits for frame buffer accesses.

Bits	Description
31:28	**A15Map**. Read-write. Reset: 0.
27:24	**A14Map**. Read-write. Reset: 0.
23:20	**A13Map**. Read-write. Reset: 0.
19:16	**A12Map**. Read-write. Reset: 0.
15:12	**A11Map**. Read-write. Reset: 0.
11:8	**A10Map**. Read-write. Reset: 0.

7:4	**A9Map**. Read-write. Reset: 0.
3:0	**A8Map**. Read-write. Reset: 0.

GMMx28[78:6C] GMC DRAM Aperture Base 3-0

Reset: 0000_0000h.

The GMC DRAM Aperture Base/Limit registers specify the portions of the frame buffer that are accessible when D18F3xF8_x4[SecureGfxMode]=1. All frame buffer accesses that do not match one of the apertures are forwarded to the 4 KB region of the frame buffer specified by GMMx2894 [GMC DRAM Aperture Default Base Address].

Bits	Description
31:20	Reserved.
19:0	**Base: frame buffer aperture base address[39:20]**. Read-write.

GMMx28[88:7C] GMC DRAM Aperture Limit 3-0

Reset: 0000_0000h.

Bits	Description
31:20	Reserved.
19:0	**Top: frame buffer aperture limit address[39:20]**. Read-write.

GMMx288C GMC C6 Save Base

Reset: 000F_FFFFh.

The GMC C6 Save Base/Limit registers specify the portion of DRAM used for saving C6 data. All accesses that match the save region are forwarded to the 4 KB region of the frame buffer specified by GMMx2894. The GMC C6 aperture is disabled when GMMx288C[Base] > GMMx2890[Top].

Bits	Description
31:20	Reserved.
19:0	**Base: C6 save base address[39:20]**. Read-write. IF ((D18F4x1AC[CoreC6Cap]==1 && D18F4x1AC[CoreC6Dis]==0) || (D18F4x1AC[PkgC6Cap]==1 && D18F4x1AC[PkgC6Dis]==0)) THEN BIOS: {D18F4x12C[C6Base],0h}. ELSE BIOS: F_FFFFh. ENDIF.

GMMx2890 GMC C6 Save Limit

Reset: 0000_0000h.

Bits	Description
31:20	Reserved.
19:0	**Top: C6 save limit address[39:20]**. Read-write. IF ((D18F4x1AC[CoreC6Cap]==1 && D18F4x1AC[CoreC6Dis]==0) || (D18F4x1AC[PkgC6Cap]==1 && D18F4x1AC[PkgC6Dis]==0)) THEN BIOS: {D18F4x12C[C6Base],Fh}. ELSE BIOS: 0_0000h. ENDIF.

GMMx2894 GMC DRAM Aperture Default Base Address

Reset: 0000_0000h.

Bits	Description
31:28	Reserved.
27:0	**Def: frame buffer aperture default base address[39:12]**. Read-write.

GMMx2898 GMC Frame Buffer Offset

Reset: 0F00_0000h.

The register along with GMMx2024 [GMC Frame Buffer Location] specifies the base address of the frame buffer in GPU physical memory and system memory.

Bits	Description
31:28	Reserved.
27:24	**Top: frame buffer region GPU limit address[23:20]**. Read-write. Specifies the GPU address of the top of frame buffer region.
23:20	**Base: frame buffer GPU base address[23:20]**. Read-write. Specifies the GPU base address of the frame buffer.
19:0	**Offset: frame buffer region system base address[39:20]**. Read-write. Specifies the system base address of frame buffer region.

GMMx5428 Configuration Memory Size

Scratch register that BIOS uses to pass the memory configuration to the GPU driver.

Bits	Description
31:0	**ConfigMemsize**. Read-write. Reset: 0. Configuration memory size.

GMMx5490 Frame Buffer Access Control

Bits	Description
31:2	Reserved.
1	**FbWriteEn**. Read-write. Reset: 0. Enables host writes to the Frame Buffer.
0	**FbReadEn**. Read-write. Reset: 0. Enables host reads to the Frame Buffer.

3.18 APIC Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention.

APIC20 APIC ID

Bits	Description
31:24	**ApicId**. Read-write. Reset: {000000b,CpuCoreNum[1:0]}. BIOS: See 2.4.6.1.1. When D18F0x68[ApicExtId and ApicExtBrdCst] = 11b, all 8 bits of this field are used; if either of these bits are low, then bits[1:0] of this field are used and bits[7:2] are reserved. See 2.4.2 [Processor Cores and Downcoring].
23:0	Reserved.

APIC30 APIC Version

Reset: 80xx_0010h.

Bits	Description
31	**ExtApicSpace: extended APIC register space present**. Read-only. This bit indicates the presence of extended APIC register space starting at APIC400.
30:24	Reserved.
23:16	**MaxLvtEntry**. Read-only. Reset: Product-specific. This field specifies the number of entries in the local vector table minus one.
15:8	Reserved.
7:0	**Version**. Read-only. This field indicates the version number of this APIC implementation.

APIC80 Task Priority

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**Priority**. Read-write. This field is assigned by software to set a threshold priority at which the core is interrupted.

APIC90 Arbitration Priority

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**Priority**. Read-only. This field indicates the current priority for a pending interrupt, or a task or interrupt being serviced by the core. The priority is used to arbitrate between cores to determine which accepts a lowest-priority interrupt request.

APICA0 Processor Priority

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**Priority**. Read-only. This field indicates the core's current priority servicing a task or interrupt, and is used to determine if any pending interrupts should be serviced. It is the higher value of the task priority value and the current highest in-service interrupt.

APICB0 End of Interrupt

This register is written by the software interrupt handler to indicate the servicing of the current interrupt is complete.

Bits	Description
31:0	Reserved. Write-only. Reads return undefined data.

APICC0 Remote Read

Reset: 0000_0000h.

Bits	Description
31:0	**RemoteReadData**. Read-only. This field contains the data resulting from a valid completion of a remote read inter-processor interrupt.

APICD0 Logical Destination

Reset: 0000_0000h.

Bits	Description
31:24	**Destination**. Read-write. This field contains this APIC's destination identification. This field is used to determine which interrupts should be accepted.
23:0	Reserved.

APICE0 Destination Format

Reset: FFFF_FFFFh.

Bits	Description
31:28	**Format**. Read-write. This field controls which format to use when accepting interrupts with a logical destination mode. The allowed values are: Bits — Definition 0h — Cluster destinations are used Fh — Flat destinations are used
27:0	Reserved.

APICF0 Spurious Interrupt Vector

Reset: 0000_00FFh.

Bits	Description
31:10	Reserved.
9	**FocusDisable**. Read-write. 1=Disable focus core checking during lowest-priority arbitrated interrupts.
8	**APICSWEn: APIC software enable**. Read-write. 0=SMI, NMI, INIT, Startup, and Remote Read interrupts may be accepted; pending interrupts in the APIC[170:100] and APIC[270:200] are held, but further fixed, lowest-priority, LINT, and ExtInt interrupts are not accepted. All LVT entry mask bits are set and cannot be cleared.
7:0	**Vector**. Read-write. This field contains the vector that is sent to the core in the event of a spurious interrupt. The behavior of bits 3:0 are controlled as specified by D18F0x68 [Link Transaction Control][ApicExtSpur].

APIC[170:100] Interrupt In-Service

Reset: 0000_0000h.

The in-service registers provide a bit per interrupt to indicate that the corresponding interrupt is being serviced by the core. APIC100[15:0] are reserved.

Register	Function
APIC100	Interrupts 31-16
APIC110	Interrupts 63-32
APIC120	Interrupts 95-64
APIC130	Interrupts 127-96
APIC140	Interrupts 159-128
APIC150	Interrupts 191-160
APIC160	Interrupts 223-192
APIC170	Interrupts 255-224

Bits	Description
31:0	**InServiceBits**. Read-only. These bits are set when the corresponding interrupt is being serviced by the core.

APIC[1F0:180] Trigger Mode

Reset: 0000_0000h.

The trigger mode registers provide a bit per interrupt to indicate that the assertion mode of each interrupt. APIC180[15:0] are reserved.

Register	Function
APIC180	Interrupts 31-16
APIC190	Interrupts 63-32

APIC1A0	Interrupts 95-64
APIC1B0	Interrupts 127-96
APIC1C0	Interrupts 159-128
APIC1D0	Interrupts 191-160
APIC1E0	Interrupts 223-192
APIC1F0	Interrupts 255-224

Bits	Description
31:0	**TriggerModeBits**. Read-only. The corresponding trigger mode bit is updated when an interrupt enters servicing. 0=Edge-triggered interrupt. 1=Level-triggered interrupt.

APIC[270:200] Interrupt Request

Reset: 0000_0000h.

The interrupt request registers provide a bit per interrupt to indicate that the corresponding interrupt has been accepted by the APIC. APIC200[15:0] are reserved.

Register	Function
APIC200	Interrupts 31-16
APIC210	Interrupts 63-32
APIC220	Interrupts 95-64
APIC230	Interrupts 127-96
APIC240	Interrupts 159-128
APIC250	Interrupts 191-160
APIC260	Interrupts 223-192
APIC270	Interrupts 255-224

Bits	Description
31:0	**RequestBits**. Read-only. The corresponding request bit is set when an interrupt is accepted by the APIC.

APIC280 Error Status

Reset: 0000_0000h.

Writes to this register trigger an update of the register state. The value written by software is arbitrary. Each write causes the internal error state to be loaded into this register, clearing the internal error state. Consequently, a second write prior to the occurrence of another error causes the register to be overwritten with cleared data.

Bits	Description
31:8	Reserved.
7	**IllegalRegAddr: illegal register address**. Read-write. This bit indicates that an access to a nonexistent register location within this APIC was attempted.

6	**RcvdIllegalVector: received illegal vector**. Read-write. This bit indicates that this APIC has received a message with an illegal vector (00h to 0Fh for fixed and lowest priority interrupts).
5	**SentIllegalVector**. Read-write. This bit indicates that this APIC attempted to send a message with an illegal vector (00h to 0Fh for fixed and lowest priority interrupts).
4	Reserved.
3	**RcvAcceptError: receive accept error**. Read-write. This bit indicates that a message received by this APIC was not accepted by this or any other APIC.
2	**SendAcceptError**. Read-write. This bit indicates that a message sent by this APIC was not accepted by any APIC.
1:0	Reserved.

APIC300 Interrupt Command Low

Reset: 0000_0000h.

Not all combinations of ICR fields are valid. Only the following combinations are valid:

Message Type	Trigger Mode	Level	Destination Shorthand
Fixed	Edge	N/A	N/A
	Level	Assert	N/A
Lowest Priority, SMI, NMI, INIT	Edge	N/A	Destination or all excluding self.
	Level	Assert	Destination or all excluding self
Startup	N/A	N/A	Destination or all excluding self

Bits	Description
31:20	Reserved.
19:18	**DestShrthnd: destination shorthand**. Read-write. This field provides a quick way to specify a destination for a message. Bits — Definition 00b — Destination field 01b — Self 10b — All including self 11b — All excluding self (This sends a message with a destination encoding of all 1s, so if lowest priority is used the message could end up being reflected back to this APIC.) If all including self or all excluding self is used, then destination mode is ignored and physical is automatically used.
17:16	**RemoteRdStat: remote read status**. Read-only. Bits — Definition 00b — Read was invalid 01b — Delivery pending 10b — Delivery done and access was valid 11b — Reserved
15	**TM: trigger mode**. Read-write. This bit indicates how this interrupt is triggered. 1=Level triggered. 0=Edge triggered.
14	**Level**. Read-write. 0=De-asserted. 1=Asserted.

13	Reserved.
12	**DlvryStat: delivery status**. Read-only. This bit is set to indicate that the interrupt has not yet been accepted by the destination core(s).
11	**DM: destination mode**. Read-write. 0=Physical. 1=Logical.
10:8	**MsgType: message type**. Read-write. Bits Definition Bits Definition 000b Fixed 100b NMI 001b Lowest Priority 101b INIT 010b SMI 110b Startup 011b Remote read 111b External interrupt
7:0	**Vector**. Read-write. This field contains the vector that is sent for this interrupt source.

APIC310 Interrupt Command High

Reset: 0000_0000h.

Bits	Description
31:24	**DestinationField**. Read-write. This field contains the destination encoding used when APIC300[DestShrthnd] is 00b.
23:0	Reserved.

APIC320 Timer Local Vector Table Entry

Reset: 0001_0000h.

Bits	Description
31:18	Reserved.
17	**Mode**. Read-write. 1=Periodic. 0=One-shot.
16	**Mask**. Read-write. If this bit is set, this local vector table entry does not generate interrupts.
15:13	Reserved.
12	**DlvryStat: delivery status**. Read-only. This bit is set to indicate that the interrupt has not yet been accepted by the core.
11	Reserved.
10:8	**MsgType: message type**. Write-only. Read always returns 000b. See 2.4.6.1.12 [Generalized Local Vector Table] for supported message types.
7:0	**Vector**. Read-write. This field contains the vector that is sent for this interrupt source.

APIC330 Thermal Local Vector Table Entry

Reset: 0001_0000h. Interrupts for this local vector table are caused by changes to the current P-state limit. This includes changes due to 2.10.3.1 [PROCHOT_L and Hardware Thermal Control (HTC)].

Bits	Description
31:17	Reserved.
16	**Mask**. Read-write. If this bit is set, this local vector table entry does not generate interrupts.

15:13	Reserved.
12	**DlvryStat: delivery status**. Read-only. This bit is set to indicate that the interrupt has not yet been accepted by the core.
11	Reserved.
10:8	**MsgType: message type**. Read-write. See 2.4.6.1.12 [Generalized Local Vector Table] for supported message types.
7:0	**Vector**. Read-write. This field contains the vector that is sent for this interrupt source.

APIC340 Performance Counter Vector Table Entry

Reset: 0001_0000h.

Interrupts for this local vector table are caused by overflows of MSRC001_00[07:04] [Performance Event Counter (PERF_CTR[3:0])].

Bits	Description
31:17	Reserved.
16	**Mask**. Read-write. If this bit is set, this local vector table entry does not generate interrupts.
15:13	Reserved.
12	**DlvryStat: delivery status**. Read-only. This bit is set to indicate that the interrupt has not yet been accepted by the core.
11	Reserved.
10:8	**MsgType: message type**. Read-write. See 2.4.6.1.12 [Generalized Local Vector Table] for supported message types.
7:0	**Vector**. Read-write. This field contains the vector that is sent for this interrupt source.

APIC350 Local Interrupt 0 (Legacy INTR) Local Vector Table Entry

Reset: 0001_0000h.

Bits	Description
31:17	Reserved.
16	**Mask**. Read-write. If this bit is set, this local vector table entry does not generate interrupts.
15	**TM: trigger mode**. Read-write. This bit indicates how this interrupt is triggered. It is defined as follows: • 0 = Edge triggered • 1 = Level triggered
14	**RmtIRR**. Read-only. If trigger mode is level, remote IRR is set when the interrupt has begun service. Remote IRR is cleared when the end of interrupt has occurred.
13	**PinPol: pin polarity**. Read-write. This bit is not used because LINT interrupts are delivered by link messages instead of individual pins.
12	**DlvryStat: delivery status**. Read-only. This bit is set to indicate that the interrupt has not yet been accepted by the core.
11	Reserved.

10:8	**MsgType: message type**. Read-write. See 2.4.6.1.12 [Generalized Local Vector Table] for supported message types.
7:0	**Vector**. Read-write. This field contains the vector that is sent for this interrupt source.

APIC360 Local Interrupt 1(Legacy NMI) Local Vector Table Entry

Reset: 0001_0000h.

Bits	Description
31:0	See: APIC350.

APIC370 Error Local Vector Table Entry

Reset: 0001_0000h.

Bits	Description
31:17	Reserved.
16	**Mask**. Read-write. If this bit is set, this local vector table entry does not generate interrupts.
15:13	Reserved.
12	**DlvryStat: delivery status**. Read-only. This bit is set to indicate that the interrupt has not yet been accepted by the core.
11	Reserved.
10:8	**MsgType: message type**. RAZ; write. See 2.4.6.1.12 [Generalized Local Vector Table] for supported message types.
7:0	**Vector**. Read-write. This field contains the vector that is sent for this interrupt source.

APIC380 Timer Initial Count

Reset: 0000_0000h.

Bits	Description
31:0	**Count**. Read-write. This field contains the value copied into the current count register when the timer is loaded or reloaded.

APIC390 Timer Current Count

Reset: 0000_0000h.

Bits	Description
31:0	**Count**. Read-only. This field contains the current value of the counter.

APIC3E0 Timer Divide Configuration

Reset: 0000_0000h.

Bits	Description
31:4	Reserved.

3	**Div[3]**. Read-write. See: Div[1:0]
2	Reserved.
1:0	**Div[1:0]**. Read-write. The Div[3],Div[1:0] are encoded as follows. Bits Divider Bits Divider 000b 2 100b 32 001b 4 101b 64 010b 8 110b 128 011b 16 111b 1

APIC400 Extended APIC Feature

Bits	Description
31:24	Reserved.
23:16	**ExtLvtCount: extended local vector table count**. Read-only. Reset: 04h. This specifies the number of extended LVT registers in the local APIC. These registers are APIC[530:500] [Extended Interrupt [3:0] Local Vector Table].
15:3	Reserved.
2	**ExtApicIdCap: extended APIC ID capable**. Read-only. Reset: 1. Indicates that the processor is capable of supporting an 8-bit APIC ID, controlled by APIC410[ExtApicIdEn].
1	**SeoiCap: specific end of interrupt capable**. Read-only. Reset: 1. This bit indicates that the APIC420 [Specific End Of Interrupt] is present.
0	**IerCap: interrupt enable register capable**. Read-only. Reset: 1. This bit indicates that the APIC[4F0:480] [Interrupt Enable] are present. See 2.4.6.1.6 [Interrupt Masking].

APIC410 Extended APIC Control

Reset: 0000_0000h.

Bits	Description
31:3	Reserved.
2	**ExtApicIdEn: extended APIC ID enable**. Read-write. 1=Enable 8-bit APIC ID; APIC20[ApicId] supports an 8-bit value; an interrupt broadcast in physical destination mode requires that the IntDest[7:0]=1111_1111b (instead of xxxx_1111b); a match in physical destination mode occurs when (IntDest[7:0] == ApicId[7:0]) instead of (IntDest[3:0] == ApicId[3:0]). Extended APIC ID can also be enabled by writing D18F0x68[ApicExtId] and D18F0x68[ApicExtBrdCst]. If this bit is set, then D18F0x68[ApicExtId] and D18F0x68[ApicExtBrdCst] must be set.
1	**SeoiEn**. Read-write. This bit enables SEOI generation when a write to the specific end of interrupt register is received.
0	**IerEn**. Read-write. This bit enables writes to the interrupt enable registers.

APIC420 Specific End Of Interrupt

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**EoiVec: end of interrupt vector**. Read-write. A write to this field causes an end of interrupt cycle to be performed for the vector specified in this field. The behavior is undefined if no interrupt is pending for the specified interrupt vector.

APIC[4F0:480] Interrupt Enable

Reset: FFFF_FFFFh.

Register	Function
APIC480	Interrupts 31-0
APIC490	Interrupts 63-32
APIC4A0	Interrupts 95-64
APIC4B0	Interrupts 127-96
APIC4C0	Interrupts 159-128
APIC4D0	Interrupts 191-160
APIC4E0	Interrupts 223-192
APIC4F0	Interrupts 255-224

Bits	Description
31:0	**InterruptEnableBits**. Read-write. The interrupt enable bits can be used to enable each of the 256 interrupts.

APIC[530:500] Extended Interrupt [3:0] Local Vector Table

Reset: 0001_0000h. These registers provide additional local vector table entries for selected internal interrupt sources. See D18F3x1CC[LvtOffset].

Bits	Description
31:17	Reserved.
16	**Mask**. Read-write. 1=This LVT entry does not generate interrupts.
15:13	Reserved.
12	**DlvryStat: delivery status**. Read-only. 1=The interrupt has not yet been accepted by the CPU.
11	Reserved.
10:8	**MsgType: message type**. Read-write. Specifies the interrupt type generated by this LVT entry. See 2.4.6.1.12 [Generalized Local Vector Table] for supported message types.
7:0	**Vector**. Read-write. This field contains the vector generated by this LVT entry.

3.19 CPUID Instruction Registers

Processor feature capabilities and configuration information are provided through the CPUID instruction. Dif-

ferent information is accessed by (1) setting EAX as an index to the registers to be read, (2) executing the CPUID instruction, and (3) reading the results in EAX, EBX, ECX, and EDX. The phrase *CPUID function X* or *CPUID FnX* refers to the CPUID instruction when EAX is preloaded with X. Undefined function numbers return 0's in all 4 registers. See 2.4.8 [CPUID Instruction].

Unless otherwise specified, the single-bit feature fields are encoded as 1=Feature is supported by the processor; 0=Feature is not supported by the processor.

The following provides AMD Family 14h Models 00h-0Fh processor specific details about CPUID. See the *CPUID Specification* for further information.

CPUID Fn0000_0000_EAX Processor Vendor and Largest Standard Function Number

Bits	Description
31:0	**LFuncStd: largest standard function**. Value: 0000_0006h. The largest CPUID standard function input value supported by the processor implementation.

CPUID Fn0000_0000_E[B,C,D]X Processor Vendor and Largest Standard Function Number

Table 102: Reset mapping for CPUID Fn0000_0000_E[B,C,D]X

Register	Value	Description
CPUID Fn0000_0000_EBX	6874_7541h	The ASCII characters: h, t, u, A
CPUID Fn0000_0000_ECX	444D_4163h	The ASCII characters: D, M, A, c
CPUID Fn0000_0000_EDX	6974_6E65h	The ASCII characters: i, t, n, e

Bits	Description
31:0	**Vendor: vendor**. The 12 8-bit ASCII character codes to create the string "AuthenticAMD".

CPUID Fn0000_0001_EAX Family, Model, Stepping Identifiers

This register provides identical information to D18F3xFC.

Family is an 8-bit value and is defined as: **Family[7:0] =** ({0000b,BaseFamily[3:0]} + ExtendedFamily[7:0]). E.g. If BaseFamily[3:0]=Fh and ExtendedFamily[7:0]=05h, then Family[7:0]=14h.

Model is an 8-bit value and is defined as: **Model[7:0]** = {ExtendedModel[3:0], BaseModel[3:0]}. E.g. If ExtendedModel[3:0]=Eh and BaseModel[3:0]=8h, then Model[7:0] = E8h. Model numbers vary by product.

This document applies only to Family 14h Models 00h-0Fh processors.

Bits	Description
31:28	Reserved.
27:20	**ExtendedFamily: extended family**. Value: 5h.
19:16	**ExtendedModel: extended model**. Value: 0h.
15:12	Reserved.
11:8	**BaseFamily: base family**. Value: Fh.

Bits	Description
7:4	**BaseModel: base model**. Value: Product-specific.
3:0	**Stepping: processor stepping (revision) for a specific model**. Value: Product-specific.

CPUID Fn0000_0001_EBX LocalApicId, LogicalProcessorCount, CLFlush, 8BitBrandId

Bits	Description
31:24	**LocalApicId: initial local APIC physical ID**. Provides the initial APIC20[ApicId] value. Changes to APIC20[ApicId] do not affect the value of this CPUID register. See 2.4.2 [Processor Cores and Downcoring].
23:16	**LogicalProcessorCount: logical processor count**. Value: Product-specific. IF (CPUID Fn0000_0001_EDX[HTT] == 1) THEN This field indicates the number of cores in the processor as CPUID Fn8000_0008_ECX[NC] + 1. ELSE Reserved. ENDIF.
15:8	**CLFlush: CLFLUSH size in quadwords**. Value: 08h.
7:0	**8BitBrandId: 8 bit brand ID**. Value: 00h. Indicates that the brand ID is in CPUID Fn8000_0001_EBX.

CPUID Fn0000_0001_ECX Feature Identifiers

Bits	Description
31:24	Reserved.
23	**POPCNT: POPCNT instruction**. Value: 1.
22:14	Reserved.
13	**CMPXCHG16B: CMPXCHG16B instruction**. Value: 1.
12:10	Reserved.
9	**SSSE3: supplemental SSE3 extensions**. Value: 1
8:4	Reserved.
3	**Monitor: Monitor/Mwait instructions**. IF (MSRC001_0015[MonMwaitDis]==0)THEN Value: 1 ELSE Reset: 0 ENDIF.
2:1	Reserved.
0	**SSE3: SSE3 extensions.** IF (MSRC001_0015[SseDis]==0)THEN Value: 1 ELSE Value: 0 ENDIF.

CPUID Fn0000_0001_EDX Feature Identifiers

Bits	Description
31:29	Reserved.
28	**HTT: hyper-threading technology**. Value: Product-specific. This bit qualifies the meaning of CPUID Fn0000_0001_EBX[LogicalProcessorCount]. 1=Multi core product (CPUID Fn8000_0008_ECX[NC] != 0). 0=Single core product (CPUID Fn8000_0008_ECX[NC] = 0).
27	Reserved.
26	**SSE2: SSE2 extensions**. IF (MSRC001_0015[SseDis]==0) THEN Value: 1 ELSE Value: 0 ENDIF.

Bits	Description
25	**SSE: SSE extensions**. IF (MSRC001_0015[SseDis]==0) THEN Value: 1 ELSE Value: 0 ENDIF.
24	**FXSR: FXSAVE and FXRSTOR instructions**. Value: 1.
23	**MMX: MMX™ instructions**. Value: 1.
22:20	Reserved.
19	**CLFSH: CLFLUSH instruction**. Value: 1.
18	Reserved.
17	**PSE36: page-size extensions**. Value: 1.
16	**PAT: page attribute table**. Value: 1.
15	**CMOV: conditional move instructions, CMOV, FCOMI, FCMOV**. Value: 1.
14	**MCA: machine check architecture, MCG_CAP**. Value: 1.
13	**PGE: page global extension, CR4.PGE**. Value: 1.
12	**MTRR: memory-type range registers**. Value: 1.
11	**SysEnterSysExit: SYSENTER and SYSEXIT instructions**. Value: 1.
10	Reserved.
9	**APIC: advanced programmable interrupt controller (APIC) exists and is enabled**. Value: MSR0000_001B[ApicEn].
8	**CMPXCHG8B: CMPXCHG8B instruction**. Value: 1.
7	**MCE: machine check exception, CR4.MCE**. Value: 1.
6	**PAE: physical-address extensions (PAE)**. Value: 1.
5	**MSR: AMD model-specific registers (MSRs), with RDMSR and WRMSR instructions**. Value: 1.
4	**TSC: time stamp counter, RDTSC/RDTSCP instructions, CR4.TSD**. Value: 1.
3	**PSE: page-size extensions (4 MB pages)**. Value: 1.
2	**DE: debugging extensions, IO breakpoints, CR4.DE**. Value: 1.
1	**VME: virtual-mode enhancements**. Value: 1.
0	**FPU: x87 floating point unit on-chip**. Value: 1.

CPUID Fn0000_000[4,3,2] Reserved

Bits	Description
31:0	Reserved.

CPUID Fn0000_0005_EAX Monitor/MWait

Bits	Description
31:16	Reserved.
15:0	**SMon: Smallest monitor-line size in bytes.** Value: 40h

CPUID Fn0000_0005_EBX Monitor/MWait

Bits	Description
31:16	Reserved.
15:0	**LMon: Largest monitor-line size in bytes.** Value: 40h.

CPUID Fn0000_0005_ECX Monitor/MWait

Bits	Description
31:2	Reserved.
1	**IBE: Interrupt break-event**. Value: 1.
0	**EMX: Enumerate MONITOR/MWAIT extensions**. Value: 1.

CPUID Fn0000_0005_EDX Monitor/MWait

Bits	Description
31:0	Reserved.

CPUID Fn0000_0006_EAX Power Management Features

Bits	Description
31:0	Reserved.

CPUID Fn0000_0006_EBX Power Management Features

Bits	Description
31:0	Reserved.

CPUID Fn0000_0006_ECX Power Management Features

Bits	Description
31:1	Reserved.
0	**EffFreq: effective frequency interface**. Value: 1. Indicates presence of MSR0000_00E7 (MPERF) and MSR0000_00E8 (APERF).

CPUID Fn0000_0006_EDX Power Management Features

Bits	Description
31:0	Reserved.

CPUID Fn8000_0000_EAX Processor Vendor and Largest Extended Function Number

Bits	Description
31:0	**LFuncExt: largest extended function**. Value: 8000_001Bh. The largest CPUID extended function input value supported by the processor implementation.

CPUID Fn8000_0000_E[B,C,D]X Processor Vendor and Largest Extended Function Number

Table 103: Reset mapping for CPUID Fn0000_0000_EAX

Register	Value	Description
CPUID Fn8000_0000_EBX	6874_7541h	The ASCII characters: h, t, u, A
CPUID Fn8000_0000_ECX	444D_4163h	The ASCII characters: D, M, A, c
CPUID Fn8000_0000_EDX	6974_6E65h	The ASCII characters: i, t, n, e

Bits	Description
31:0	**Vendor: vendor**. The 12 8-bit ASCII character codes to create the string “AuthenticAMD”.

CPUID Fn8000_0001_EAX Family, Model, Stepping Identifiers

See CPUID Fn0000_0001_EAX.

Bits	Description
31:28	Reserved.
27:20	**ExtendedFamily: extended family**. Value: 05h.
19:16	**ExtendedModel: extended model**. Value: 0h.
15:12	Reserved.
11:8	**BaseFamily: base family**. Value: Fh.
7:4	**BaseModel: base model**. Value: Product-specific.
3:0	**Stepping: processor stepping (revision) for a specific model**. Value: Product-specific.

CPUID Fn8000_0001_EBX BrandId Identifier

Bits	Description
31:28	**PkgType: package type**. Value: Product-specific. Specifies the package type. This field is encoded as follows: Bits — Description 0000b — FT1 (BGA)
27:16	Reserved.
15:0	**BrandId: brand ID**. Value: D18F3x1F0[BrandId].

CPUID Fn8000_0001_ECX Feature Identifiers

Bits	Description
31:14	Reserved.
13	**WDT: watchdog timer support**. Value: 1.
12	**SKINIT: SKINIT and STGI support**. Value: 1.
11	**SSE5: instruction support**. Value: 0.
10	**IBS: Instruction Based Sampling**. Value: 1.
9	**OSVW: OS Visible Work-around support**. Value: 1.
8	**3DNowPrefetch: Prefetch and PrefetchW instructions**. Value: 1.
7	**MisAlignSse: Misaligned SSE Mode.** IF (MSRC001_0015[MisAlignSseDis]==0) THEN Value: 1 ELSE Value: 0 ENDIF.
6	**SSE4A: EXTRQ, INSERTQ, MOVNTSS, and MOVNTSD instruction support.** IF (MSRC001_0015[SseDis]==0) THEN Value: 1 ELSE Value: 0 ENDIF.
5	**ABM: advanced bit manipulation**. Value: 1. LZCNT instruction support.
4	**AltMovCr8: LOCK MOV CR0 means MOV CR8**. Value: 1.
3	**ExtApicSpace: extended APIC register space**. Value: 1.
2	**SVM: Secure Virtual Mode feature**. Value: Product-specific. Indicates support for: VMRUN, VMLOAD, VMSAVE, CLGI, VMMCALL, and INVLPGA.
1	**CmpLegacy: core multi-processing legacy mode**. Value: Product-specific. 1=Multi core product (CPUID Fn8000_0008_ECX[NC] != 0). 0=Single core product (CPUID Fn8000_0008_ECX[NC] = 0).
0	**LahfSahf: LAHF/SAHF instructions**. Value: 1.

CPUID Fn8000_0001_EDX Feature Identifiers

Bits	Description
31	**3DNow: 3DNow!™ instructions**. Value: 0.
30	**3DNowExt: AMD extensions to 3DNow!™ instructions**. Value: 0.
29	**LM: long mode**. Value: Product-specific.
28	Reserved.
27	**RDTSCP: RDTSCP instruction**. Value: 1.
26	**Page1GB: one GB large page support**. Value: 1.
25	**FFXSR: FXSAVE and FXRSTOR instruction optimizations**. Value: 1.
24	**FXSR: FXSAVE and FXRSTOR instructions**. Value: 1.
23	**MMX: MMX™ instructions**. Value: 1.
22	**MmxExt: AMD extensions to MMX™ instructions.** Value: 1.
21	Reserved.
20	**NX: no-execute page protection**. Value: 1.

Bits	Description
19:18	Reserved.
17	**PSE36: page-size extensions**. Value: 1.
16	**PAT: page attribute table**. Value: 1.
15	**CMOV: conditional move instructions, CMOV, FCOMI, FCMOV**. Value: 1.
14	**MCA: machine check architecture, MCG_CAP**. Value: 1.
13	**PGE: page global extension, CR4.PGE**. Value: 1.
12	**MTRR: memory-type range registers**. Value: 1.
11	**SysCallSysRet: SYSCALL and SYSRET instructions**. Value: 1.
10	Reserved.
9	**APIC: advanced programmable interrupt controller (APIC) exists and is enabled**. Value: MSR0000_001B[ApicEn].
8	**CMPXCHG8B: CMPXCHG8B instruction**. Value: 1.
7	**MCE: machine check exception, CR4.MCE**. Value: 1.
6	**PAE: physical-address extensions (PAE)**. Value: 1.
5	**MSR: AMD model-specific registers (MSRs), with RDMSR and WRMSR instructions**. Value: 1.
4	**TSC: time stamp counter, RDTSC/RDTSCP instructions, CR4.TSD**. Value:1.
3	**PSE: page-size extensions (4 MB pages)**. Value: 1.
2	**DE: debugging extensions, IO breakpoints, CR4.DE**. Value: 1.
1	**VME: virtual-mode enhancements**. Value: 1.
0	**FPU: x87 floating point unit on-chip**. Value: 1.

CPUID Fn8000_000[4,3,2]_E[D,C,B,A]X Processor Name String Identifier

Table 104: Reset mapping for CPUID Fn8000_000[4,3,2]_E[D,C,B,A]X

Register	Value
CPUID Fn8000_0002_EAX	MSRC001_0030[31:0]
CPUID Fn8000_0002_EBX	MSRC001_0030[63:32]
CPUID Fn8000_0002_ECX	MSRC001_0031[31:0]
CPUID Fn8000_0002_EDX	MSRC001_0031[63:32]
CPUID Fn8000_0003_EAX	MSRC001_0032[31:0]
CPUID Fn8000_0003_EBX	MSRC001_0032[63:32]
CPUID Fn8000_0003_ECX	MSRC001_0033[31:0]
CPUID Fn8000_0003_EDX	MSRC001_0033[63:32]
CPUID Fn8000_0004_EAX	MSRC001_0034[31:0]
CPUID Fn8000_0004_EBX	MSRC001_0034[63:32]
CPUID Fn8000_0004_ECX	MSRC001_0035[31:0]
CPUID Fn8000_0004_EDX	MSRC001_0035[63:32]

Bits	Description
31:0	**ProcName: processor name**. These return the ASCII string corresponding to the processor name, stored in MSRC001_00[35:30] [Processor Name String].

CPUID Fn8000_0005_EAX L1 Cache and TLB Identifiers

This provides the processor's first level cache and TLB characteristics for each core. The *associativity* fields returned are encoded as follows:

00h Reserved.
01h Direct mapped.
02h - FEh Specifies the associativity; e.g., 04h would indicate a 4-way associativity.
FFh - Fully associative.

Bits	Description
31:24	**L1DTlb2and4MAssoc: data TLB associativity for 2 MB and 4 MB pages**. Value: FFh.
23:16	**L1DTlb2and4MSize: data TLB number of entries for 2 MB and 4 MB pages**. Value: 08h. The value returned is for the number of entries available for the 2 MB page size; 4 MB pages require two 2 MB entries, so the number of entries available for the 4 MB page size is one-half the returned value.
15:8	**L1ITlb2and4MAssoc: instruction TLB associativity for 2 MB and 4 MB pages**. Value: FFh.
7:0	**L1ITlb2and4MSize: instruction TLB number of entries for 2 MB and 4 MB pages**. Value: 08h. The value returned is for the number of entries available for the 2 MB page size; 4 MB pages require two 2 MB entries, so the number of entries available for the 4 MB page size is one-half the returned value.

CPUID Fn8000_0005_EBX L1 Cache and TLB Identifiers

See CPUID Fn8000_0005_EAX.

Bits	Description
31:24	**L1DTlb4KAssoc: data TLB associativity for 4 KB pages**. Value: FFh.
23:16	**L1DTlb4KSize: data TLB number of entries for 4 KB pages**. Value: 40.
15:8	**L1ITlb4KAssoc: instruction TLB associativity for 4 KB pages**. Value: 00h. ITLB associativity for 4 KB pages is reported by CPUID Fn8000_0006_EBX[L2ITlb4KAssoc].
7:0	**L1ITlb4KSize: instruction TLB number of entries for 4 KB pages**. Value: 0. ITLB size for 4 KB pages is reported by CPUID Fn8000_0006_EBX[L2ITlb4KSize].

CPUID Fn8000_0005_ECX L1 Cache and TLB Identifiers

See CPUID Fn8000_0005_EAX.

Bits	Description
31:24	**L1DcSize: L1 data cache size in KB**. Value: 32.
23:16	**L1DcAssoc: L1 data cache associativity**. Value: 8.
15:8	**L1DcLinesPerTag: L1 data cache lines per tag**. Value: 1.
7:0	**L1DcLineSize: L1 data cache line size in bytes**. Value: 64.

CPUID Fn8000_0005_EDX L1 Cache and TLB Identifiers

See CPUID Fn8000_0005_EAX.

Bits	Description
31:24	**L1IcSize: L1 instruction cache size KB**. Value: 32.
23:16	**L1IcAssoc: L1 instruction cache associativity**. Value: 2.
15:8	**L1IcLinesPerTag: L1 instruction cache lines per tag**. Value: 1.
7:0	**L1IcLineSize: L1 instruction cache line size in bytes**. Value: 64.

CPUID Fn8000_0006_EAX L2 Cache and L2 TLB Identifiers

This provides the processor's second level cache and TLB characteristics for each core.

The presence of a unified L2 TLB is indicated by a value of 0000h in the upper 16 bits of the EAX and EBX registers. The unified L2 TLB information is contained in the lower 16 bits of these registers.

The *associativity* fields are encoded as follows:

0h: The L2 cache or TLB is disabled.
1h: Direct mapped.
2h: 2-way associative.
4h: 4-way associative.
6h: 8-way associative.
8h: 16-way associative.
Ah: 32-way associative.
Bh: 48-way associative.
Ch: 64-way associative.
Dh: 96-way associative.
Eh: 128-way associative.
Fh: Fully associative.

All other encodings are reserved.

Bits	Description
31:28	**L2DTlb2and4MAssoc: L2 data TLB associativity for 2-MB and 4-MB pages**. Value: 0.
27:16	**L2DTlb2and4MSize: L2 data TLB number of entries for 2-MB and 4-MB pages**. Value: 0. The value returned is for the number of entries available for the 2-MB page size; 4-MB pages require two 2-MB entries, so the number of entries available for the 4-MB page size is one-half the returned value.
15:12	**L2ITlb2and4MAssoc: L2 instruction TLB associativity for 2-MB and 4-MB pages**. Value: 0.
11:0	**L2ITlb2and4MSize: L2 instruction TLB number of entries for 2-MB and 4-MB pages**. Value: 0. The value returned is for the number of entries available for the 2-MB page size; 4-MB pages require two 2-MB entries, so the number of entries available for the 4-MB page size is one-half the returned value.

CPUID Fn8000_0006_EBX L2 Cache and L2 TLB Identifiers

See CPUID Fn8000_0006_EAX.

Bits	Description
31:28	**L2DTlb4KAssoc: L2 data TLB associativity for 4 KB pages**. Value: 4.
27:16	**L2DTlb4KSize: L2 data TLB number of entries for 4 KB pages**. Value: 512.
15:12	**L2ITlb4KAssoc: L2 instruction TLB associativity for 4 KB pages**. Value: 4.
11:0	**L2ITlb4KSize: L2 instruction TLB number of entries for 4 KB pages**. Value: 512.

CPUID Fn8000_0006_ECX L2 Cache and L2 TLB Identifiers

See CPUID Fn8000_0006_EAX.

Bits	Description
31:16	**L2Size: L2 cache size in KB**. Value: 0200h.
15:12	**L2Assoc: L2 cache associativity**. Value: 1000b. 1000b=16-way associative cache.
11:8	**L2LinesPerTag: L2 cache lines per tag**. Value: 1.
7:0	**L2LineSize: L2 cache line size in bytes**. Value: 64.

CPUID Fn8000_0006_EDX L2 Cache and L2 TLB Identifiers

Bits	Description
31:0	Reserved.

CPUID Fn8000_0007_E[A,B,C]X Advanced Power Management Information

Bits	Description
31:0	Reserved.

CPUID Fn8000_0007_EDX Advanced Power Management Information

This function provides advanced power management feature identifiers.

Bits	Description
31:11	Reserved.
10	**TdpLimits: software and APML TDP limiting is supported**. Value: 0. TDP limiting is not supported.
9	**CPB: core performance boost is supported**. Value: 0.
8	**TscInvariant: TSC invariant**. Value: 1. The TSC rate is invariant.
7	**HwPstate: hardware P-state control is supported**. Value: 1. MSRC001_0061 [P-State Current Limit], MSRC001_0062 [P-State Control] and MSRC001_0063 [P-State Status] exist.
6	**100MHzSteps: 100 MHz multiplier Control**. Value: 1.
5	Reserved.
4	**TM: hardware thermal control (HTC) is supported**. Value: Product-specific.
3	**TTP: THERMTRIP is supported**. Value: 1.
2	**VID: Voltage ID control is supported**. Value: 0. Function replaced by HwPstate.
1	**FID: Frequency ID control is supported**. Value: 0. Function replaced by HwPstate.
0	**TS: Temperature sensor**. Value: 1.

CPUID Fn8000_0008_EAX Long Mode Address Size Identifiers

This provides information about the maximum physical and linear address width supported by the processor.

Bits	Description
31:16	Reserved.
15:8	**LinAddrSize: Maximum linear byte address size in bits**. IF (CPUID Fn8000_0001_EDX[LM]==1) THEN Value: 30h. ELSE Value: 20h. ENDIF.
7:0	**PhysAddrSize: Maximum physical byte address size in bits**. Value: 24h.

CPUID Fn8000_0008_EBX Long Mode Address Size Identifiers

Bits	Description
31:0	Reserved.

CPUID Fn8000_0008_ECX Long Mode Address Size Identifiers

This provides information about the number of cores supported by the processor.

Bits	Description
31:16	Reserved.
15:12	**ApicIdCoreIdSize: APIC ID size**. Value: 1h. The number of bits in the initial APIC20[ApicId] value that indicate core ID within a processor.
11:8	Reserved.
7:0	**NC: number of physical cores - 1**. Value: Product-specific. The number of cores in the processor is NC+1 (e.g., if NC=0, then there is one core). See 2.4.2 [Processor Cores and Downcoring].

CPUID Fn8000_0008_EDX Long Mode Address Size Identifiers

Bits	Description
31:0	Reserved.

CPUID Fn8000_0009 Reserved

Bits	Description
31:0	Reserved.

IF (CPUID Fn8000_0001_ECX[SVM]==1) THEN.

CPUID Fn8000_000A_EAX SVM Revision and Feature Identification

Provides SVM revision and feature information.

Bits	Description
31:8	Reserved.
7:0	**SvmRev: SVM revision**. Value: 01h.

ENDIF.

IF (CPUID Fn8000_0001_ECX[SVM]==1) THEN.

CPUID Fn8000_000A_EBX SVM Revision and Feature Identification

Provides SVM revision and feature information.

Bits	Description
31:0	**NASID: number of address space identifiers (ASID)**. Value: 8h.

ENDIF.

CPUID Fn8000_000A_ECX SVM Revision and Feature Identification

Bits	Description
31:0	Reserved.

IF (CPUID Fn8000_0001_ECX[SVM]==1) THEN.

CPUID Fn8000_000A_EDX SVM Revision and Feature Identification

Provides SVM revision and feature information.

Bits	Description
31:11	Reserved.
10	**PauseFilter: pause intercept filter**. Value: 1.
9:5	Reserved.
4	**TscRateMsr: MSR based TSC rate control**. Value: 0.
3	**NRIPS: NRIP Save**. Value: 1. Indicates support for NRIP save on #VMEXIT.
2	**SVML: SVM lock**. Value: 1.
1	**LbrVirt: LBR virtualization**. Value: 1.
0	**NP: nested paging**. Value: 1.

ENDIF.

CPUID Fn8000_00[18:0B] Reserved

Bits	Description
31:0	Reserved.

CPUID Fn8000_0019_EAX TLB 1GB Page Identifiers

This provides 1 GB paging information. The *associativity* fields are defined by CPUID Fn8000_0006_EAX, CPUID Fn8000_0006_EBX, CPUID Fn8000_0006_ECX and CPUID Fn8000_0006_EDX.

Bits	Description
31:28	**L1DTlb1GAssoc: L1 data TLB associativity for 1 GB pages**. Value: 0.
27:16	**L1DTlb1GSize: L1 data TLB number of entries for 1 GB pages**. Value: 0.
15:12	**L1ITlb1GAssoc: L1 instruction TLB associativity for 1 GB pages**. Value: 0.
11:0	**L1ITlb1GSize: L1 instruction TLB number of entries for 1 GB pages**. Value: 0.

CPUID Fn8000_0019_EBX TLB 1GB Page Identifiers

This provides 1 GB paging information. The *associativity* fields are defined by CPUID Fn8000_0006_EAX, CPUID Fn8000_0006_EBX, CPUID Fn8000_0006_ECX and CPUID Fn8000_0006_EDX.

Bits	Description
31:28	**L2DTlb1GAssoc: L2 data TLB associativity for 1 GB pages**. Value: 0.
27:16	**L2DTlb1GSize: L2 data TLB number of entries for 1 GB pages**. Value: 0.
15:12	**L2ITlb1GAssoc: L2 instruction TLB associativity for 1 GB pages**. Value: 0.
11:0	**L2ITlb1GSize: L2 instruction TLB number of entries for 1 GB pages**. Value: 0.

CPUID Fn8000_0019_E[C,D]X TLB 1GB Page Identifiers

Bits	Description
31:0	Reserved.

CPUID Fn8000_001A_EAX Performance Optimization Identifiers

This function returns performance related information.

Bits	Description
31:2	Reserved.
1	**MOVU: movu**. Value: 0.
0	**FP128: fp128**. Value: 0.

CPUID Fn8000_001A_E[B,C,D]X Performance Optimization Identifiers

Bits	Description
31:0	Reserved.

CPUID Fn8000_001B_EAX Instruction Based Sampling Identifiers

This function returns IBS feature information.

Bits	Description
31:8	Reserved.
7	**RipInvalidChk: invalid RIP indication supported**. RIP Invalid is supported in MSRC001_1035[IbsRipInvalid]. Value: 1.
6	**OpCntExt: IbsOpCurCnt and IbsOpMaxCnt extend by 7 bits**. Value: 1.
5	**BrnTrgt: branch target address reporting supported**. Value: 1.
4	**OpCnt: op counting mode supported**. Value: 1.
3	**RdWrOpCnt: read write of op counter supported**. Value: 1.
2	**OpSam: IBS execution sampling supported**. Value: 1.
1	**FetchSam: IBS fetch sampling supported**. Value: 1.
0	**IBSFFV: IBS feature flags valid**. Value: 1.

CPUID Fn8000_001B_E[B,C,D]X Instruction Based Sampling Identifiers

Bits	Description
31:0	Reserved.

3.20 MSRs - MSR0000_xxxx

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. MSRs are accessed through x86 WRMSR and RDMSR instructions.

MSR0000_0000 Load-Store MCA Address

Bits	Description
63:0	Alias of MSR0000_0402.

MSR0000_0001 Load-Store MCA Status

Bits	Description
63:0	Alias of MSR0000_0401.

MSR0000_0010 Time Stamp Counter (TSC)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**TSC: time stamp counter**. Read-write. After reset, this register increments as defined by MSRC001_0015[TscFreqSel]. The TSC counts at the same rate in all P-states, all C states, S0, or S1.

MSR0000_001B APIC Base Address (APIC_BAR)

Bits	Description
63:36	MBZ.
35:12	**ApicBar[35:12]: APIC base address register**. Read-write. Reset: 0_FEE0_0h. Specifies the base address for the APICXX register set, memory mapped to a 4 KB range. The base address of this range is specified by {MSR0000_001B[ApicBar[35:12], 000h}. See 3.18 [APIC Registers] for details about this register set.
11	**ApicEn: APIC enable**. Read-write. Reset: 0. 1=Local APIC enabled; the APICXX register set is accessible; all interrupt types are accepted. 0=Local APIC disabled; the APICXX register set is not accessible; only non-vectored interrupts are supported including NMI, SMI, INIT and ExtINT; local-vector-table interrupts can still occur if the LVTs have been previously programmed.
10:9	MBZ.
8	**BSC: boot strap core**. Read-write. Reset: xb. 1=The core is the BSC. 0=The core is not the BSC.
7:0	MBZ.

MSR0000_002A Cluster ID (EBL_CR_POWERON)

Reset: 0000_0000_0000_0000h. Read; GP-write. Attempted writes to this register result in general protection faults with error code 0.

Bits	Description
63:18	Reserved.
17:16	**ClusterID: APIC cluster ID**. This is normally 00b; the value does not affect hardware.
15:0	Reserved.

MSR0000_00E7 Max Performance Frequency Clock Count (MPERF)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**MPERF: maximum core clocks counter**. Read-write. This field is incremented by hardware at the P0 frequency while the core is in the C0 state. In combination with MSR0000_00E8, this is used to determine the effective frequency of the core. See MSRC001_0015[EffFreqCntMwait], and 2.5.3.3 [Effective Frequency].

MSR0000_00E8 Actual Performance Frequency Clock Count (APERF)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**APERF: actual core clocks counter**. Read-write. This field is incremented by hardware for each core clock cycle that occurs while the core is in the C0 state. In combination with MSR0000_00E7, this is used to determine the effective frequency of the core. See MSRC001_0015[EffFreqCntMwait] and 2.5.3.3 [Effective Frequency].

MSR0000_00FE MTRR Capabilities (MTRRcap)

Reset: 0000_0000_0000_0508h. Read; GP-write.

Bits	Description
63:11	Reserved.
10	**MtrrCapWc: write-combining memory type**. 1=The write combining memory type is supported.
9	Reserved.
8	**MtrrCapFix: fixed range register**. 1=Fixed MTRRs are supported.
7:0	**MtrrCapVCnt: variable range registers count**. Specifies the number of variable MTRRs supported.

MSR0000_0174 SYSENTER CS (SYSENTER_CS)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:32	RAZ.

31:16	Reserved.
15:0	**SYSENTER_CS: SYSENTER target CS**. Read-write. Holds the called procedure code segment.

MSR0000_0175 SYSENTER ESP (SYSENTER_ESP)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:32	RAZ.
31:0	**SYSENTER_ESP: SYSENTER target SP**. Read-write. Holds the called procedure stack pointer.

MSR0000_0176 SYSENTER EIP (SYSENTER_EIP)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:32	Reserved.
31:0	**SYSENTER_EIP: SYSENTER target IP**. Read-write. Holds the called procedure instruction pointer.

MSR0000_0179 Global Machine Check Capabilities (MCG_CAP)

Read; GP-write.

Bits	Description
63:9	Reserved.
8	**MCG_CTL_P: MCG_CTL register present**. Value: 1. 1=The machine check control registers (MCi_CTL; see 2.16 [Machine Check Architecture]) are present.
7:0	**Count**. Value: 06h. Indicates the number of error-reporting banks supported. 06h=Error-reporting banks 0 through 5. See 2.16.1 [Machine Check Registers].

MSR0000_017A Global Machine Check Status (MCG_STAT)

Reset: 0000_0000_0000_0000h. See 2.16 [Machine Check Architecture].

Bits	Description
63:3	Reserved.
2	**MCIP: machine check in progress**. Read-write; set-by-hardware. 1=A machine check is in progress.
1	**EIPV: error instruction pointer valid**. Read-write; updated-by-hardware. 1=The instruction pointer that was pushed onto the stack by the machine check mechanism references the instruction that caused the machine check error.
0	**RIPV: restart instruction pointer valid**. Read-write; updated-by-hardware. 1=Program execution can be reliably restarted at the EIP address on the stack.

MSR0000_017B Global Machine Check Exception Reporting Control (MCG_CTL)

Reset: 0000_0000_0000_0000h.

This register enables the various machine check register banks. See 2.16 [Machine Check Architecture]. It is expected that this register is programmed to the same value in all cores.
When a machine check register bank is disabled, errors for that bank are not detected or logged.

Bits	Description
63:6	Reserved.
5	**MC5En: reorder buffer register bank enable**. Read-write. 1=The fixed-issue reorder buffer (FR) machine check register bank is enabled.
4	**MC4En: northbridge register bank enable**. Read-write. 1=The northbridge (NB) machine check register bank is enabled.
3	**MC3En: MC3 register bank enable**. RAZ. Not used.
2	**MC2En: bus unit register bank enable**. Read-write. 1=The bus unit (BU) machine check register bank is enabled.
1	**MC1En: instruction cache register bank enable**. Read-write. 1=The instruction cache (IC) machine check register bank is enabled.
0	**MC0En: data cache register bank enable**. Read-write. 1=The data cache (DC) machine check register bank is enabled.

MSR0000_01D9 Debug Control (DBG_CTL_MSR)

Reset: 0000_0000_0000_0000h. For more information about this MSR, see the APM2 section titled “Debug-Control MSR (DebugCtlMSR)”.

Bits	Description
63:7	Reserved.
6	MBZ.
5:2	**PB: performance monitor pin control**. Read-write. This field does not control any hardware.
1	**BTF**. Read-write. 1=Enable branch single step.
0	**LBR**. Read-write. 1=Enable last branch record.

MSR0000_01DB Last Branch From IP (BR_FROM)

Reset: 0000_0000_0000_0000h. Read; GP-write. For more information about this MSR, see the APM2 section titled “Control-Transfer Recording MSRs”.

Bits	Description
63:0	**LastBranchFromIP**. Loaded with the segment offset of the branch instruction.

MSR0000_01DC Last Branch To IP (BR_TO)

Reset: 0000_0000_0000_0000h. Read; GP-write. For more information about this MSR, see the APM2 section titled “Control-Transfer Recording MSRs”.

Bits	Description
63:0	**LastBranchToIP**. Holds the target RIP of the last branch that occurred before an exception or interrupt.

MSR0000_01DD Last Exception From IP

Reset: 0000_0000_0000_0000h. Read; GP-write. For more information about this MSR, see the APM2 section titled “Control-Transfer Recording MSRs”.

Bits	Description
63:0	**LastIntFromIP.** Holds the source RIP of the last branch that occurred before the exception or interrupt.

MSR0000_01DE Last Exception To IP

Reset: 0000_0000_0000_0000h. Read; GP-write. For more information about this MSR, see the APM2 section titled “Control-Transfer Recording MSRs”.

Bits	Description
63:0	**LastIntToIP**. Holds the target RIP of the last branch that occurred before the exception or interrupt.

MSR0000_020[E,C,A,8,6,4,2,0] Variable-Size MTRRs (MTRRphysBase*n*)

Each MTRR (MSR0000_020[E,C,A,8,6,4,2,0] [Variable-Size MTRRs (MTRRphysBasen)], MSR0000_02[6F:68,59,58,50] [Fixed-Size MTRRs], or MSR0000_02FF [MTRR Default Memory Type (MTRRdefType)]) specifies a physical address range and a corresponding memory type (MemType) associated with that range. Setting the memory type to an unsupported value results in a #GP(0).

The variable-size MTRRs come in pairs of base and mask registers (MSR0000_0200 and MSR0000_0201 are the first pair, etc.). Variables MTRRs are enabled through MSR0000_02FF [MTRR Default Memory Type (MTRRdefType)][MtrrDefTypeEn]. A CPU access--with address CPUAddr--is determined to be within the address range of a variable-size MTRR if the following equation is true:

CPUAddr[35:12] & PhyMask[35:12] == PhyBase[35:12] & PhyMask[35:12].

For example, if the variable MTRR spans 256K bytes and starts at the 1M byte address. The PhyBase would be set to 0_0010_0000h and the PhyMask to F_FFFC_0000h (with zeros filling in for bits[11:0]). This results in a range from 0_0010_0000h to 0_0013_FFFF.

Bits	Description
63:36	MBZ.
35:12	**PhyBase: base address**. Read-write. Reset: 0.

11:8	MBZ.
7:3	Reserved.
2:0	**MemType: memory type**. Read-write. Reset: 0. Bits Definition Bits Definition 0h Uncacheable (UC) 4h Write through (WT) 1h Write combining (WC) 5h Write protect (WP) 2h Reserved 6h Write back (WB) 3h Reserved 7h Reserved

MSR0000_020[F,D,B,9,7,5,3,1] Variable-Size MTRRs (MTRRphysMask*n*)

See MSR0000_020[E,C,A,8,6,4,2,0].

Bits	Description
63:36	MBZ.
35:12	**PhyMask: address mask**. Read-write. Reset: 0.
11	**Valid: valid**. Read-write. 1=The variable-size MTRR pair is enabled. Reset: 0.
10:0	MBZ.

MSR0000_02[6F:68,59,58,50] Fixed-Size MTRRs

See MSR0000_020[E,C,A,8,6,4,2,0] for general MTRR information. Fixed MTRRs are enabled through MSR0000_02FF[MtrrDefTypeFixEn and MtrrDefTypeEn].

Mapping Table 1: Fixed-size MTRR size and range mapping

Register	Bits							
	63:56	55:48	47:40	39:32	31:24	23:16	15:8	7:0
MSR0000_0250	64K_70000	64K_60000	64K_50000	64K_40000	64K_30000	64K_20000	64K_10000	64K_00000
MSR0000_0258	16K_9C000	16K_98000	16K_94000	16K_90000	16K_8C000	16K_88000	16K_84000	16K_80000
MSR0000_0259	16K_BC000	16K_B8000	16K_B4000	16K_B0000	16K_AC000	16K_A8000	16K_A4000	16K_A0000
MSR0000_0268	4K_C7000	4K_C6000	4K_C5000	4K_C4000	4K_C3000	4K_C2000	4K_C1000	4K_C0000
MSR0000_0269	4K_CF000	4K_CE000	4K_CD000	4K_CC000	4K_CB000	4K_CA000	4K_C9000	4K_C8000
MSR0000_026A	4K_D7000	4K_D6000	4K_D5000	4K_D4000	4K_D3000	4K_D2000	4K_D1000	4K_D0000
MSR0000_026B	4K_DF000	4K_DE000	4K_DD000	4K_DC000	4K_DB000	4K_DA000	4K_D9000	4K_D8000
MSR0000_026C	4K_E7000	4K_E6000	4K_E5000	4K_E4000	4K_E3000	4K_E2000	4K_E1000	4K_E0000
MSR0000_026D	4K_EF000	4K_EE000	4K_ED000	4K_EC000	4K_EB000	4K_EA000	4K_E9000	4K_E8000
MSR0000_026E	4K_F7000	4K_F6000	4K_F5000	4K_F4000	4K_F3000	4K_F2000	4K_F1000	4K_F0000
MSR0000_026F	4K_FF000	4K_FE000	4K_FD000	4K_FC000	4K_FB000	4K_FA000	4K_F9000	4K_F8000

Bits	Description
63:61	MBZ.
60	**RdDram: read DRAM**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][4].
59	**WrDram: write DRAM**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][3].
58:56	**MemType: memory type**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][2:0].

55:53	MBZ.
52	**RdDram: read DRAM**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][4].
51	**WrDram: write DRAM**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][3].
50:48	**MemType: memory type**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][2:0].
47:45	MBZ.
44	**RdDram: read DRAM**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][4].
43	**WrDram: write DRAM**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][3].
42:40	**MemType: memory type**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][2:0].
39:37	MBZ.
36	**RdDram: read DRAM**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][4].
35	**WrDram: write DRAM**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][3].
34:32	**MemType: memory type**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][2:0].
31:29	MBZ.
28	**RdDram: read DRAM**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][4].
27	**WrDram: write DRAM**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][3].
26:24	**MemType: memory type**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][2:0].
23:21	MBZ.
20	**RdDram: read DRAM**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][4].
19	**WrDram: write DRAM**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][3].
18:16	**MemType: memory type**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][2:0].
15:13	MBZ.
12	**RdDram: read DRAM**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][4].
11	**WrDram: write DRAM**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][3].
10:8	**MemType: memory type**. Read-write. Reset: 0. See: MSR0000_02[6F:68,59,58,50][2:0].
7:5	MBZ.
4	**RdDram: read DRAM**. IF (MSRC001_0010[MtrrFixDramModEn]) THEN Read-write. ELSE MBZ. ENDIF. Reset: 0. 0=Read accesses to the range are marked as MMIO. 1=Read accesses to the range are marked as destined for DRAM. See MSRC001_0010[MtrrFixDramEn, MtrrFixDramModEn].
3	**WrDram: write DRAM**. IF (MSRC001_0010[MtrrFixDramModEn]) THEN Read-write. ELSE MBZ. ENDIF. Reset: 0. 0=Write accesses to the range are marked as MMIO. 1=Write accesses to the range are marked as destined for DRAM. See MSRC001_0010[MtrrFixDramEn, MtrrFixDram-ModEn].
2:0	**MemType: memory type**. Read-write. Reset: 0. Bits Definition — Bits Definition 0h Uncacheable (UC) — 4h Write through (WT) 1h Write combining (WC) — 5h Write protect (WP) 2h Reserved — 6h Write back (WB) 3h Reserved — 7h Reserved

MSR0000_0277 Page Attribute Table (PAT)

Reset: 0007_0406_0007_0406h.

This register specifies the memory type based on the PAT, PCD, and PWT bits in the virtual address page tables. The encodings for PA[7:0] is:

0h = UC or uncacheable.
1h = WC or write combining.
4h = WT or write through.
All other values result in a #GP(0).
5h = WP or write protect.
6h = WB or write back.
7h = UC- or uncacheable (overridden by MTRR WC state)

Bits	Description
63:59	MBZ.
58:56	**PA7MemType**. Read-write. Default UC. MemType for {PAT, PCD, PWT} = 7h.
55:51	MBZ.
50:48	**PA6MemType**. Read-write. Default UC-. MemType for {PAT, PCD, PWT} = 6h.
47:43	MBZ.
42:40	**PA5MemType**. Read-write. Default WT. MemType for {PAT, PCD, PWT} = 5h.
39:35	MBZ.
34:32	**PA4MemType**. Read-write. Default WB. MemType for {PAT, PCD, PWT} = 4h.
31:27	MBZ.
26:24	**PA3MemType**. Read-write. Default UC. MemType for {PAT, PCD, PWT} = 3h.
23:19	MBZ.
18:16	**PA2MemType**. Read-write. Default UC-. MemType for {PAT, PCD, PWT} = 2h.
15:11	MBZ.
10:8	**PA1MemType**. Read-write. Default WT. MemType for {PAT, PCD, PWT} = 1h.
7:3	MBZ.
2:0	**PA0MemType**. Read-write. Default WB. MemType for {PAT, PCD, PWT} = 0h.

MSR0000_02FF MTRR Default Memory Type (MTRRdefType)

Reset: 0000_0000_0000_0000h.

See MSR0000_020[E,C,A,8,6,4,2,0] and MSR0000_020[F,D,B,9,7,5,3,1] for general MTRR information.

Bits	Description
63:12	MBZ.
11	**MtrrDefTypeEn: variable and fixed MTRR enable**. Read-write. 1=MSR0000_020[E,C,A,8,6,4,2,0] and MSR0000_020[F,D,B,9,7,5,3,1], and MSR0000_02[6F:68,59,58,50] are enabled. 0=Fixed and variable MTRRs are not enabled.
10	**MtrrDefTypeFixEn: fixed MTRR enable**. Read-write. 1=MSR0000_02[6F:68,59,58,50] are enabled. This field is ignored (and the fixed MTRRs are not enabled) if MSR0000_02FF[MtrrDefTypeEn]=0.
9:8	MBZ.

7:0	**MemType: memory type**. Read-write. The memory type for memory space that is not specified by either the fixed or variable range MTRR's is defined as a function of MtrrDefTypeEn as follows: • If MtrrDefTypeEn==1 then the default memory type is MemType. • If MtrrDefTypeEn==0 then the default memory type is UC.

MSR0000_0400 DC Machine Check Control (MC0_CTL)

Reset: 0000_0000_0000_0000h.
BIOS: FFFF_FFFF_FFFF_FFFFh. Because some legacy operating systems do not properly initialize this register, BIOS must initialize this register. See 2.16 [Machine Check Architecture]. For all bits, 1=Enable the specified reporting mechanism.

Bits	Description
63:12	Reserved.
11	**SRDE_ALL: all system read data errors**. Read-write. Report system read data errors for any operation including a DC/IC fetch, TLB reload.
10	**SRDET: read data errors on TLB reload**. Read-write. Report system read data errors on a TLB reload if MSR0000_0400[SRDE_ALL] = 1.
9	**SRDES: read data errors on store**. Read-write. Report system read data errors on a store if MSR0000_0400[SRDE_ALL] = 1.
8	**SRDEL: read data errors on load**. Read-write. Report system read data errors on a load if MSR0000_0400[SRDE_ALL] = 1.
7	Reserved.
6	**TLBP: TLB parity errors**. Read-write. Report TLB parity errors.
5:4	Reserved.
3	**DTP: tag array parity errors**. Read-write. Report data cache tag array parity errors.
2	**DDP: data array parity errors**. Read-write. Report data cache data array parity errors.
1:0	Reserved.

MSR0000_0401 DC Machine Check Status (MC0_STATUS)

Cold reset: 0000_0000_0000_0000h.

See 2.16 [Machine Check Architecture]. Each of the MCi_STATUS registers hold information identifying the last error logged in each bank. Software is normally only allowed to write 0's to these registers to clear the fields so subsequent errors may be logged. See MSRC001_0015[McStatusWrEn]. The following field definitions apply to all MCi_STATUS registers, except as noted.

Bits	Description
63	**Val: error valid**. Read-write; set-by-hardware. 1=This bit indicates that a valid error has been detected. This bit should be cleared to 0 by software after the register has been read.
62	**Over: error overflow**. Read-write; set-by-hardware. 1=An error was detected while the valid bit (Val) was set; at least one error was not logged. Overflow is set independently of whether the existing error is overwritten. See 2.16.2.2 [Machine Check Error Logging Overwrite During Overflow].
61	**UC: error uncorrected**. Read-write; set-by-hardware. 1=The error was not corrected by hardware.

60	**En: error enable**. Read-write; set-by-hardware. 1=MCA error reporting is enabled for this error in MC0_CTL.
59	Reserved.
58	**AddrV: error address valid**. Read-write; updated-by-hardware. 1=The address saved in MC0_ADDR is the address where the error occurred.
57	**PCC: processor context corrupt**. Read-write; updated-by-hardware. 1=The state of the processor may have been corrupted by the error condition. Restart may not be reliable.
56:20	Reserved.
19:16	**ErrorCodeExt[3:0]: extended error code**. Read-write; updated-by-hardware. See Table 106.
15:0	**ErrorCode: error code**. Read-write; updated-by-hardware. See Table 105 and Table 106.

Table 105: DC error descriptions

Error Type	Description	Enablers MSR0000_0400
Data or Tag Parity	A DC data or tag array parity error for a tag hit occurred during an LS access to the DC.	DDP, DTP
Copyback Parity	A DC data or tag array parity error for a tag miss occurred during an LS access to the DC.	DDP, DTP
Tag Snoop Parity	A tag parity error was encountered during snoop. Data array parity check enabled only for tag hit.	DDP, DTP
L2TLB Parity	Parity error in BTLB.	TLBP
L2TLB Multimatch	Hit multiple entries.	TLBP
Read Data Error on TLB Reload	System read data error occurred on a TLB reload.	SRDET
Read Data Error on Store	System read data error occurred on a store.	SRDES
Read Data Error on Load	System read data error occurred on a load.	SRDEL

Table 106: DC error signatures

Error Type	**[19:16] Error-CodeExt**	**ErrorCode (see 2.16.2.3)**						**[61] UC**	**[58] ADDRV**	**[57] PCC**
		Type	**10:9 PP**	**8 TT**	**7:4 RRRR**	**3:2 II/TT**	**1:0 LL**			
Data or Tag Parity due to Load or HW Prefetch	0000	MEM	-	-	DRD	Data	L1	1	1	1
Data or Tag Parity due to Store	0000	MEM	-	-	DWR	Data	L1	1	1	1
Copyback Par-ity	0000	MEM	-	-	Evict	Data	L1	1	1	1
Tag Snoop Parity	0000	MEM	-	-	Snoop	Data	L1	1	1	1
L2TLB Parity	0000	TLB	-	-	-	Data	L2	1	1	1

Table 106: DC error signatures

Error Type	[19:16] Error-CodeExt	ErrorCode (see 2.16.2.3)						[61] UC	[58] ADDRV	[57] PCC
		Type	10:9 PP	8 TT	7:4 RRRR	3:2 II/TT	1:0 LL			
L1TLB Multi-match	0001	TLB	-	-	-	Data	L1	1	1	1
L2TLB Multi-match	0001	TLB	-	-	-	Data	L2	1	1	1
Read Data Error on TLB Reload	0000	BUS	SRC	0	RD	MEM/IO	LG	1	1	0
Read Data Error on Store	0000	BUS	SRC	0	DWR	MEM	LG	1	1	1
Read Data Error on Load	0000	BUS	SRC	0	DRD	MEM/IO	LG	1	1	1

MSR0000_0402 DC Machine Check Address (MC0_ADDR)

Cold reset: 0000_0000_0000_0000h.

See 2.16 [Machine Check Architecture]. Each of the MCi_ADDR registers are written to by hardware and read-write accessible by software. MCi_ADDR registers contains valid data if indicated by MCi_STATUS[AddrV]. Table 107 defines the address register as a function of error type.

Bits	Description
63:48	Reserved.
47:0	**ADDR**. Read-write; updated-by-hardware. See Table 107.

Table 107: DC error data; address register

Error Type	Address Register Bits	Description
Data or Tag Parity due to HW Prefetch	35:6	Physical address
Data or Tag Parity due to Load or Store	35:4	Physical address
Copyback Parity	11:6	Physical address
Tag Snoop Parity	35:6	Physical address
BTLB Parity	47:12	Linear address
BTLB Multi-match		
Read Data Error on TLB Reload, Load, or Store.	47:3	Physical address

MSR0000_0403 DC Machine Check Miscellaneous (MC0_MISC)

Cold reset: 0000_0000_0000_0000h.

Bits	Description
63:0	Reserved.

MSR0000_0404 IC Machine Check Control (MC1_CTL)

Reset: 0000_0000_0000_0000h. See 2.16 [Machine Check Architecture]. For all bits, 1=Enable the specified reporting mechanism.

Bits	Description
63:10	Reserved.
9	**SRDE: read data errors**. Read-write. Report system read data errors for an instruction cache fetch if MSR0000_0400[SRDE_ALL] = 1.
8:7	Reserved.
6	**TLBP: TLB parity errors**. Read-write. Report TLB parity errors.
5	Reserved.
4	**ISTP: snoop tag array parity errors**. Read-write. Report instruction cache snoop tag array parity errors which occur during snoop.
3	**ITP: tag array parity errors**. Read-write. Report instruction cache tag array parity errors.
2	**IDP: data array parity errors**. Read-write. Report instruction cache data array parity errors.
1:0	Reserved.

MSR0000_0405 IC Machine Check Status (MC1_STATUS)

Cold reset: 0000_0000_0000_0000h.

See 2.16 [Machine Check Architecture] and MSRC001_0015[McStatusWrEn].

Bits	Description
63	**Val: error valid**. Read-write; set-by-hardware. 1=This bit indicates that a valid error has been detected. This bit should be cleared to 0 by software after the register has been read.
62	**Over: error overflow**. Read-write; set-by-hardware. 1=An error was detected while the valid bit (Val) was set; at least one error was not logged. Overflow is set independently of whether the existing error is overwritten. See 2.16.2.2 [Machine Check Error Logging Overwrite During Overflow].
61	**UC: error uncorrected**. Read-write; set-by-hardware. 1=The error was not corrected by hardware.
60	**En: error enable**. Read-write; set-by-hardware. 1=MCA error reporting is enabled for this error in MC1_CTL.
59	Reserved.
58	**AddrV: error address valid**. Read-write; updated-by-hardware. 1=The address saved in MC1_ADDR is the address where the error occurred.
57	**PCC: processor context corrupt**. Read-write; updated-by-hardware. 1=The state of the processor may have been corrupted by the error condition. Restart may not be reliable.

56:20	Reserved.
19:16	**ErrorCodeExt[3:0]: extended error code**. Read-write; updated-by-hardware. See Table 109.
15:0	**ErrorCode: error code**. Read-write; updated-by-hardware. See Table 109.

This register reports these IC errors:

Table 108: IC error descriptions

Error Type	Description	Enablers MSR0000_0404
Read Data Error	An error occurred during an attempted read of data from the NB. Possible reasons include master abort and target abort.	SRDE
Data or Tag Parity	An IC data or tag array parity error for a tag hit occurred during instruction fetch from the IC. The data is discarded from the IC, the data array and all tag arrays are cleared, and the line is refetched. IC parity errors may occur on non-cacheable instructions.	IDP, ITP
Tag Snoop	A tag error was encountered during snoop or victimization.	ISTP
BTLB Parity	Parity error in BTLB.	TLBP
BTLB Multimatch	Hit multiple entries.	TLBP

Table 109: IC error signatures

Error Type	[19:16] Error-CodeExt	ErrorCode (see 2.16.2.3)						[61] UC	[58] ADDRV	[57] PCC
		Type	10:9 PP	8 TT	7:4 RRRR	3:2 II/TT	1:0 LL			
Read Data Error	0000	BUS	SRC	0	IRD	MEM	LG	1[3]	0	0
Data or Tag Parity	0000	MEM	-	-	IRD	Instr	L1	0[1]	1	0
Tag Snoop	0000	MEM	-	-	Snoop	Instr	L1	1[3]	1	1
BTLB Parity	0000	TLB	-	-	-	Instr	L1	0[2]	1	0
BTLB Multimatch	0001	TLB	-	-	-	Instr	L1	0[2]	1	0

1. Recoverable, microcode flushes the Instruction Cache and refetches from memory.
2. Recoverable, microcode flushes TLB and refetches from memory.
3. Unrecoverable, Machine Check taken.

MSR0000_0406 IC Machine Check Address (MC1_ADDR)

Cold reset: 0000_0000_0000_0000h.

See 2.16 [Machine Check Architecture]. Each of the MCi_ADDR registers are written to by hardware and read-write accessible by software. MCi_ADDR registers contains valid data if indicated by

MCi_STATUS[AddrV]. Table 110 defines the address register as a function of error type

Bits	Description
63:48	Reserved.
47:0	**ADDR**. Read-write; updated-by-hardware. See Table 110.

The following table defines the address register as a function of error type.

Table 110: IC error data; address register

Error Type	Address Register Bits	Description
IC Data and Tag Parity	47:3	Linear address
Tag Snoop	35:6	Physical address
BTLB Parity	47:3	Linear address
BTLB Multimatch		

MSR0000_0407 IC Machine Check Miscellaneous (MC1_MISC)

Cold reset: 0000_0000_0000_0000h.

Bits	Description
63:0	Reserved.

MSR0000_0408 BU Machine Check Control (MC2_CTL)

Reset: 0000_0000_0000_0000h.

See 2.16 [Machine Check Architecture]. For all bits, 1=Enable the specified reporting mechanism. A given error will always update the STATUS register unless the corresponding MASK bit in MC2_CTL_MASK is set.

Bits	Description
63:14	Reserved.
13	**AttrParity: report attribute parity errors.** Read-write.
12	**TagMultiHit: report tag multi hit errors.** Read-write.
11:9	Reserved.
8	**DataUncor: report uncorrectable data errors.** Read-write.
7	**DataCor: report correctable data errors.** Read-write.
6	**DataParity: report data parity errors.** Read-write.
5	**TagUncor: report uncorrectable tag errors.** Read-write.
4	**TagCor: report correctable tag errors.** Read-write.
3:0	Reserved.

MSR0000_0409 BU Machine Check Status (MC2_STATUS)

Cold reset: 0000_0000_0000_0000h.

Bits	Description
63	**Val: error valid**. Read-write; set-by-hardware. 1=This bit indicates that a valid error has been detected. This bit should be cleared to 0 by software after the register has been read.
62	**Over: error overflow**. Read-write; set-by-hardware. 1=An error was detected while the valid bit (Val) was set; at least one error was not logged. Overflow is set independently of whether the existing error is overwritten. See 2.16.2.2 [Machine Check Error Logging Overwrite During Overflow].
61	**UC: error uncorrected**. Read-write; set-by-hardware. 1=The error was not corrected by hardware.
60	**En: error enable**. Read-write; set-by-hardware. 1=MCA error reporting is enabled for this error in MC2_CTL.
59	Reserved.
58	**AddrV: error address valid**. Read-write; updated-by-hardware. 1=The address saved in MC2_ADDR is the address where the error occurred.
57	**PCC: processor context corrupt**. Read-write; updated-by-hardware. 1=The state of the processor may have been corrupted by the error condition. Restart may not be reliable.
56:47	Reserved.
46	**CECC: correctable ECC error**. Read-write; updated-by-hardware. 1=The error was a correctable ECC error.
45	**UECC: uncorrectable ECC error**. Read-write; updated-by-hardware. 1=The error was an uncorrectable ECC error.
44:21	Reserved.
20:16	**ErrorCodeExt:** Extended error code. Read-write; updated-by-hardware. See Table 111
15:0	**ErrorCode: error code**. Read-write; updated-by-hardware. See Table 111.

Table 111: BU error signatures

Error Type	Access Type	[20:16] ErrorCodeExt	Error Code (see D18F3x48 for encoding)			[61] UC	[58] ADDRV	[57] PCC	[46] CECC	[45] UECC
			15:8 Type	7:4 Memory	1:0 LL					
Data	Instr Fetch	00000b	Memory	IRD	L2	1/0	1	If UC	1/0	1/0
	Data Fetch	00000b	Memory	DRD	L2	1/0	1	If UC	1/0	1/0
	Snoop	00000b	Memory	Snoop	L2	1/0	1	If UC	1/0	1/0
	Evict	00000b	Memory	Evict	L2	1/0	1	If UC	1/0	1/0
Tag	Instr Fetch	00010b	Memory	IRD	L2	1/0	1	If UC	1/0	1/0
	Data Fetch	00010b	Memory	DRD	L2	1/0	1	If UC	1/0	1/0
	Snoop	00010b	Memory	Snoop	L2	1	1	1	0	1/0
	Evict	00010b	Memory	Evict	L2	1	1	1	0	1/0
	Scrub	00010b	Memory	Gen	L2	1/0	1	If UC	1/0	1/0

MSR0000_040A BU Machine Check Address (MC2_ADDR)

Cold reset: 0000_000x_xxxx_xxxxh. See 2.16 [Machine Check Architecture]. The following table defines the

address register as a function of error type.

Bits	Description
63:36	Reserved.
35:0	**ADDR**. Read-write; updated-by-hardware. See Table 112.

Table 112: BU error data; address register

Error Type	Address Register Bits	Description
System Data Read Error	35:6	Physical address
L2 Cache Data		
Tag	3:0	Encoded L2 way
	14:6	L2 Index

MSR0000_040B BU Machine Check Miscellaneous (MC2_MISC)

Cold reset: 0000_0000_0000_0000h.

Bits	Description
63:0	Reserved.

MSR0000_040C MC3 Machine Check Control (MC3_CTL)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	RAZ.

MSR0000_040D MC3 Machine Check Status (MC3_STATUS)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	RAZ.

MSR0000_040E MC3 Machine Check Address (MC3_ADDR)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	RAZ.

MSR0000_040F MC3 Machine Check Miscellaneous (MC3_MISC)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	RAZ.

MSR0000_0410 NB Machine Check Control (MC4_CTL)

Reset: xxxx_xxxx_xxxx_xxxxh. This register is also accessible through PCI configuration space; see D18F3x40 [MCA NB Control] for layout and reset information.

Only one of these registers exists in multi-core devices; See 3.1.1 [Northbridge MSRs In Multi-Core Products].

The corresponding MCi_CTL_MASK register is MSRC001_0048 (see MSRC001_00[49:44]).

Bits	Description
63:32	Reserved.
31:0	See: D18F3x40.

MSR0000_0411 NB Machine Check Status (MC4_STATUS)

Reset: xxxx_xxxx_xxxx_xxxxh. This register is also accessible through PCI configuration space as registers D18F3x4C and D18F3x48; see those registers for layout and reset information. See 2.16 [Machine Check Architecture] for an overview of the MCA registers.

Table 94 lists and describes each error type logged. Table 95 and Table 96 describe the error codes and status register settings for each error type logged.

Only one of these registers exists in multi-core devices; see 3.1.1 [Northbridge MSRs In Multi-Core Products].

Bits	Description
63:32	See: D18F3x4C.
31:0	See: D18F3x48.

MSR0000_0412 NB Machine Check Address (MC4_ADDR)

Cold reset: xxxx_xxxx_xxxx_xxxxh. Read-write. This register is also accessible through PCI configuration space as registers D18F3x54 and D18F3x50; see those registers for layout and reset information.

Only one of these registers exists in multi-core devices; see 3.1.1 [Northbridge MSRs In Multi-Core Products].

Bits	Description
63:32	See: D18F3x54.
31:0	See: D18F3x50.

MSR0000_0413 NB Machine Check Miscellaneous (MC4_MISC)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	Reserved.

MSR0000_0414 FR Machine Check Control (MC5_CTL)

Reset: 0000_0000_0000_0000h. See 2.16 [Machine Check Architecture]. For all bits, 1=Enable the specified reporting mechanism.

Bits	Description
63:1	Reserved.
0	**CPUWDT: CPU watchdog timer**. Read-write. The core WDT expiration (see MSRC001_0074 [CPU Watchdog Timer (CpuWdTmrCfg)]).

MSR0000_0415 FR Machine Check Status (MC5_STATUS)

Cold reset: 0000_0000_0000_0000h.

Bits	Description
63	**Val: error valid**. Read-write; set-by-hardware. 1=This bit indicates that a valid error has been detected. This bit should be cleared to 0 by software after the register has been read.
62	**Over: error overflow**. Read-write; set-by-hardware. 1=An error was detected while the valid bit (Val) was set; at least one error was not logged. Overflow is set independently of whether the existing error is overwritten. See 2.16.2.2 [Machine Check Error Logging Overwrite During Overflow].
61	**UC: error uncorrected**. Read-write; set-by-hardware. 1=The error was not corrected by hardware.
60	**En: error enable**. Read-write; set-by-hardware. 1=MCA error reporting is enabled for this error in MC5_CTL.
59	**MiscV: miscellaneous error register valid**. Read-only. 1=MC5_MISC contains valid information for this error.
58	**AddrV: error address valid**. Read-write; updated-by-hardware. 1=The address saved in MC5_ADDR is the address where the error occurred.
57	**PCC: processor context corrupt**. Read-write; updated-by-hardware. 1=The state of the processor may have been corrupted by the error condition. Restart may not be reliable.
56:16	Reserved.
15:0	**ErrorCode: error code**. Read-write. See Table 114.

This register reports these FR errors:

Table 113: FR error descriptions

Error Type	Description	Enablers MSR0000_0414
CPU watchdog timer expire	See MSRC001_0074 [CPU Watchdog Timer (CpuWdTmrCfg)].	CPUWDT

Table 114: FR error signatures

Error Type	[9:16] Error-CodeExt	ErrorCode (see 2.16.2.3)						[61] UC	[58] ADDRV	[57] PCC	[46] CECC	[45] UECC
		Type	10:9 PP	8 TT	7:4 RRRR	3:2 II/TT	1:0 LL					
CPU watchdog timer expire	----	Bus	Gen	1	GEN	Gen	LG	1	1	1	0	0

MSR0000_0416 FR Machine Check Address (MC5_ADDR)

Cold reset: 0000_0000_0000_0000h. Read-write.

Bits	Description
63:49	Reserved.
48:0	**ADDR**. Read-write; updated-by-hardware. See Table 115.

The following tables define the address register as a function of error type.

Table 115: FR error data; address register

Error Type	Address Register Bits	Description
CPU watchdog timer expire	48:0	Logical address

MSR0000_0417 FR Machine Check Miscellaneous (MC5_MISC)

Cold reset: 0000_0000_0000_0000h. This register records unspecified, implementation-specific status bits when an FR machine check error is logged.

Bits	Description
63:8	Reserved.
7:0	**FrCompl.** Read-write.

3.21 MSRs - MSRC000_0xxx

MSRC000_0080 Extended Feature Enable (EFER)

Reset: 0000_0000_0000_0000h.

SKINIT Execution: 0000_0000_0000_0000h.

Bits	Description
63:15	MBZ.
14	**FFXSE: fast FXSAVE/FRSTOR enable**. Read-write. 1=Enables the fast FXSAVE/FRSTOR mechanism. A 64-bit operating system uses CPUID Fn0000_0001_EDX[FXSR] to determine the presence of this feature before enabling it. This bit is set once by the operating system and its value is not changed afterwards.
13	**LMSLE: long mode segment limit enable**. Read-write. 1=Enables the long mode segment limit check mechanism.
12	**SVME: secure virtual machine (SVM) enable**. Read-write. 1=SVM features are enabled.
11	**NXE: no-execute page enable**. Read-write. 1=The no-execute page protection feature is enabled.
10	**LMA: long mode active**. Read-only. 1=Indicates that long mode is active.
9	MBZ.
8	**LME: long mode enable**. Read-write. 1=Long mode is enabled.
7:1	RAZ.
0	**SYSCALL: system call extension enable**. Read-write. 1=SYSCALL and SYSRET instructions are enabled. This adds the SYSCALL and SYSRET instructions which can be used in flat addressed operating systems as low latency system calls and returns.

MSRC000_0081 SYSCALL Target Address (STAR)

Reset: 0000_0000_0000_0000h. This register holds the target address used by the SYSCALL instruction and the code and stack segment selector bases used by the SYSCALL and SYSRET instructions.

Bits	Description
63:48	**SysRetSel: SYSRET CS and SS**. Read-write.
47:32	**SysCallSel: SYSCALL CS and SS**. Read-write.
31:0	**Target: SYSCALL target address**. Read-write.

MSRC000_0082 Long Mode SYSCALL Target Address (STAR64)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**LSTAR: long mode target address**. Read-write. Target address for 64-bit mode calling programs. The address stored in this register must be in canonical form (if not canonical, a #GP fault occurs).

MSRC000_0083 Compatibility Mode SYSCALL Target Address (STARCOMPAT)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**CSTAR: compatibility mode target address**. Read-write. Target address for compatibility mode. The address stored in this register must be in canonical form (if not canonical, a #GP fault occurs).

MSRC000_0084 SYSCALL Flag Mask (SYSCALL_FLAG_MASK)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:32	RAZ.
31:0	**MASK: SYSCALL flag mask**. Read-write. This register holds the EFLAGS mask used by the SYSCALL instruction. 1=Clear the corresponding EFLAGS bit when executing the SYSCALL instruction.

MSRC000_0100 FS Base (FS_BASE)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**FS_BASE: expanded FS segment base**. Read-write. This register provides access to the expanded 64-bit FS segment base. The address stored in this register must be in canonical form (if not canonical, a #GP fault fill occurs).

MSRC000_0101 GS Base (GS_BASE)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**GS_BASE: expanded GS segment base**. Read-write. This register provides access to the expanded 64-bit GS segment base. The address stored in this register must be in canonical form (if not canonical, a #GP fault fill occurs).

MSRC000_0102 Kernel GS Base (KernelGSbase)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**KernelGSBase: kernel data structure pointer**. Read-write. This register holds the kernel data structure pointer which can be swapped with the GS_BASE register using the SwapGS instruction. The address stored in this register must be in canonical form (if not canonical, a #GP fault occurs).

MSRC000_0103 Auxiliary Time Stamp Counter (TSC_AUX)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:32	Reserved.
31:0	**TscAux: auxiliary time stamp counter data**. Read-write. It is expected that this is initialized by privileged software to a meaningful value, such as a processor ID. This value is returned in the RDTSCP instruction.

3.22 MSRs - MSRC001_0xxx

MSRC001_00[03:00] Performance Event Select (PERF_CTL[3:0])

Reset: 0000_0000_0000_0000h.

These registers are used to specify the events counted by MSRC001_00[07:04] [Performance Event Counter (PERF_CTR[3:0])] and to control other aspects of their operation. Each performance counter supported has a corresponding event-select register that controls its operation. Section 3.24.1 [CPU Performance Counter Events] shows the events and unit masks supported by the processor, and Section 3.24.2 [NB Performance Counter Events] shows the events and unit masks supported by the Northbridge.

To accurately start counting with the write that enables the counter, disable the counter when changing the event and then enable the counter with a second MSR write.

The edge count mode increments the counter when a transition happens on the monitored event. If the event selected is changed without disabling the counter, an extra edge is falsely detected when the first event is a static 0 and the second event is a static one. To avoid this false edge detection, disable the counter when changing the event and then enable the counter with a second MSR write.

The performance counter registers can be used to track events in the Northbridge. Northbridge events include all memory controller events and crossbar events documented in 3.24.2.1 and 3.24.2.2. Monitoring of Northbridge events should only be performed by one core. If a Northbridge event is selected using one of the Performance Event-Select registers in any core of a multi-core processor, then a Northbridge performance event cannot be selected in the same Performance Event Select register of any other core.

Care must be taken when measuring Northbridge or other non-processor-specific events under conditions where the processor may go into halt mode during the measurement period. For instance, one may wish to monitor DRAM traffic due to DMA activity from a disk or graphics adaptor. This entails running some event counter monitoring code on the processor, where such code accesses the counters at the beginning and end of the measurement period, or may even sample them periodically throughout the measurement period. Such code typically gives up the processor during each measurement interval. If there is nothing else for the OS to run on that particular processor at that time, it may halt the processor until it is needed. Under these circumstances, the clock for the counter logic may be stopped, hence the counters would not count the events of interest. To prevent this, simply run a low-priority background process that keeps the processor busy during the period of

interest.

Bits	Description
63:42	Reserved.
41	**HostOnly: host only counter**. Read-write. 1=Events are only counted when the processor is in host mode.
40	**GuestOnly: guest only counter**. Read-write. 1=Events are only counted when the processor is in guest mode.
39:36	Reserved.
35:32	**EventSelect[11:8]: performance event select**. Read-write. See: EventSelect[7:0].
31:24	**CntMask: counter mask**. Read-write. Controls the number of events counted per clock cycle. Bits — Definition 00h — The corresponding PERF_CTR[3:0] register is incremented by the number of events occurring in a clock cycle. Maximum number of events in one cycle is 3. 03h-01h — When Inv = 0, the corresponding PERF_CTR[3:0] register is incremented by 1 if the number of events occurring in a clock cycle is greater than or equal to the CntMask value. When Inv = 1, the corresponding PERF_CTR[3:0] register is incremented by 1 if the number of events occurring in a clock cycle is less than CntMask value. FFh-04h — Reserved.
23	**Inv: invert counter mask**. Read-write. See CntMask.
22	**En: enable performance counter**. Read-write. 1= Performance event counter is enabled.
21	Reserved.
20	**Int: enable APIC interrupt**. Read-write. 1=APIC performance counter LVT interrupt is enabled to generate an interrupt when the performance counter overflows.
19	Reserved.
18	**Edge: edge detect**. Read-write. 0=Level detect. 1=Edge detect.
17	**OS: OS mode**. Read-write. 1=Events are only counted when CPL=0.
16	**User: user mode**. Read-write. 1=Events only counted when CPL>0.
15:8	**UnitMask: event qualification**. Read-write. Each UnitMask bit further specifies or qualifies the event specified by EventSelect. All events selected by UnitMask are simultaneously monitored. Unless otherwise stated, the UnitMask values shown may be combined (logically ORed) to select any desired combination of the sub-events for a given event. In some cases, certain combinations can result in misleading counts, or the UnitMask value is an ordinal rather than a bit mask. These situations are described where applicable, or should be obvious from the event descriptions. For events where no UnitMask table is shown, the UnitMask is not applicable and may be set to zeros.
7:0	**EventSelect[7:0]: event select.** Read-write. This field, along with EventSelect[11:8] above, combine to form the 12-bit event select field, EventSelect[11:0]. EventSelect specifies the event or event duration in a processor unit to be counted by the corresponding PERF_CNT[3:0] register. The events are specified in section 3.24 [Performance Counter Events]. Some events are reserved; when a reserved event is selected, the results are undefined.

MSRC001_00[07:04] Performance Event Counter (PERF_CTR[3:0])

Reset: 0000_0000_0000_0000h.

The core provides four 48-bit performance counters. Each counter can monitor a different event specified by MSRC001_00[03:00] [Performance Event Select (PERF_CTL[3:0])]. The accuracy of the counters is not ensured.

Performance counters are used to count specific processor events, such as data-cache misses, or the duration of events, such as the number of clocks it takes to return data from memory after a cache miss. During event counting, the processor increments the counter when it detects an occurrence of the event. During duration measurement, the processor counts the number of processor clocks it takes to complete an event. Each performance counter can be used to count one event, or measure the duration of one event at a time.

In addition to the RDMSR instruction, the PERF_CNT[3:0] registers can be read using a special read performance-monitoring counter instruction, RDPMC. The RDPMC instruction loads the contents of the PERF_CTR[3:0] register specified by the ECX register, into the EDX register and the EAX register.

Writing the performance counters can be useful if there is an intention for software to count a specific number of events, and then trigger an interrupt when that count is reached. An interrupt can be triggered when a performance counter overflows. Software should use the WRMSR instruction to load the count as a two's-complement negative number into the performance counter. This causes the counter to overflow after counting the appropriate number of times.

The performance counters are not assured of producing identical measurements each time they are used to measure a particular instruction sequence, and they should not be used to take measurements of very small instruction sequences. The RDPMC instruction is not serializing, and it can be executed out-of-order with respect to other instructions around it. Even when bound by serializing instructions, the system environment at the time the instruction is executed can cause events to be counted before the counter value is loaded into EDX:EAX.

Bits	Description
63:48	RAZ.
47:0	**CTR: performance counter value**. Read-write. Returns the current value of the event counter.

MSRC001_0010 System Configuration (SYS_CFG)

Reset: 0000_0000_0002_0000h.

Bits	Description
63:23	Reserved.
22	**Tom2ForceMemTypeWB: top of memory 2 memory type write back**. Read-write. 1=The default memory type of memory between 4GB and TOM2 is write back instead of the memory type defined by MSR0000_02FF [MTRR Default Memory Type (MTRRdefType)][MemType]. For this bit to have any effect, MSR0000_02FF[MtrrDefTypeEn] must be 1. MTRRs and PAT can be used to override this memory type.
21	**MtrrTom2En: MTRR top of memory 2 enable**. Read-write. 0=MSRC001_001D is disabled. 1=MSRC001_001D is enabled.

20	**MtrrVarDramEn: MTRR variable DRAM enable**. Read-write. BIOS: 1. 0=MSRC001_001A [Top Of Memory (TOP_MEM)] and IORRs are disabled. 1=These registers are enabled.
19	**MtrrFixDramModEn: MTRR fixed RdDram and WrDram modification enable**. Read-write. 0=Reads of MSR0000_02[6F:68,59,58,50][RdDram, WrDram] return 0 and non-zero writes to MSR0000_02[6F:68,59,58,50][RdDram, WrDram] generate a GP exception. 1=MSR0000_02[6F:68,59,58,50][RdDram, WrDram] are read-write. BIOS: This bit should be set to 1 during BIOS initialization of the fixed MTRRs, then cleared to 0 for operation.
18	**MtrrFixDramEn: MTRR fixed RdDram and WrDram attributes enable**. Read-write. BIOS: 1. 1=Enables the RdDram and WrDram attributes in MSR0000_02[6F:68,59,58,50].
17	**SysUcLockEn: system lock command enable**. Read-write. 1=Transactions on CCI support the lock command. This is normally enabled in multi-core systems and disabled in single core systems.
16:0	Reserved.

MSRC001_0015 Hardware Configuration (HWCR)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:27	Reserved.
26	**EffFreqCntMwait: effective frequency counting during MWAIT**. Read-write. Specifies whether MSR0000_00E7 [Max Performance Frequency Clock Count (MPERF)] and MSR0000_00E8 [Actual Performance Frequency Clock Count (APERF)] increment while the core is in the monitor event pending state. 0=The registers do not increment. 1=The registers increment. See 2.5.3.3 [Effective Frequency].
25	Reserved.
24	**TscFreqSel: TSC frequency select**. Read-write. BIOS: 1. Specifies the rate at which the TSC increments. 0=The TSC increments at the main PLL frequency (see D18F3xD4[MainPllOpFreqId]). 1=The TSC increments at the frequency defined by P0 at the time this bit is set by software. Changing the state of this bit after setting it to 1 results in undefined behavior from the TSC. Changing the CPU COFs defined by MSRC001_00[6B:64] after setting this bit has no effect on the TSC rate.
23	**ForceUsRdWrSzPrb: force probes for upstream RdSized and WrSized**. Read-write; per-node. BIOS: See 2.9.3.5. 1=Forces probes on all upstream read-sized and write-sized transactions. This bit is shared between all processor cores.
22	Reserved.
21	**MisAlignSseDis: misaligned SSE mode disable**. Read-write. 1=Disables misaligned SSE mode. If this is set, then CPUID Fn8000_0001_ECX[MisAlignSse] is 0.
20	**IoCfgGpFault: IO-space configuration causes a GP fault**. Read-write. 1=IO-space accesses to configuration space cause a GP fault. The fault is triggered if any part of the IO read/write address range is between CF8h and CFFh, inclusive. These faults only result from single IO instructions, not to string and REP IO instructions. This fault takes priority over the IO trap mechanism described by MSRC001_0054 [IO Trap Control (SMI_ON_IO_TRAP_CTL_STS)].
19	Reserved.

18	**McStatusWrEn: machine check status write enable**. Read-write. 1=Writes by software to MCi_STATUS (see 2.16 [Machine Check Architecture]) do not cause general protection faults; such writes update all implemented bits in these registers. 0=Writing a non-zero pattern to these registers causes a general protection fault. McStatusWrEn can be used to debug machine check interrupt handlers. When McStatusWrEn is set, privileged software can write non-zero values to the specified registers without generating exceptions, and then simulate a machine check using the "int 18" instruction. Setting a reserved bit in these registers does not generate an exception when this mode is enabled. However, setting a reserved bit may result in undefined behavior.
17	**Wrap32Dis: 32-bit address wrap disable**. Read-write. 1=Disable 32-bit address wrapping. Software can use Wrap32Dis to access physical memory above 4 Gbytes without switching into 64-bit mode. To do so, software should write a greater-than 4 Gbyte address to MSRC000_0100 [FS Base (FS_BASE)] and MSRC000_0101 [GS Base (GS_BASE)]. Then it would address ±2 Gbytes from one of those bases using normal memory reference instructions with a FS or GS override prefix. However, the INVLPG, FST, and SSE store instructions generate 32-bit addresses in legacy mode, regardless of the state of Wrap32Dis.
16	Reserved.
15	**SseDis: SSE instructions disable**. Read-write. 1=Disables SSE instructions. If this is set, then CPUID Fn8000_0001_EDX[SSE, SSE2], CPUID Fn0000_0001_ECX[SSE3], and CPUID Fn8000_0001_ECX[SSE4A] are 0.
14	**RsmSpCycDis: RSM special bus cycle disable**. IF (MSRC001_0015[SmmLock]==1) THEN Read-only ELSE Read-write. ENDIF. 0=A link special bus cycle, SMIACK, is generated on a resume from SMI.
13	**SmiSpCycDis: SMI special bus cycle disable**. IF (MSRC001_0015[SmmLock]==1) THEN Read-only ELSE Read-write. ENDIF. 0=A link special bus cycle, SMIACK, is generated when an SMI interrupt is taken.
12:11	Reserved.
10	**MonMwaitUserEn: MONITOR/MWAIT user mode enable**. Read-write. 1=The MONITOR and MWAIT instructions are supported in all privilege levels. 0=The MONITOR and MWAIT instructions are supported only in privilege level 0; these instructions in privilege levels 1 to 3 cause a #UD exception. The state of this bit is ignored if MonMwaitDis is set.
9	**MonMwaitDis: MONITOR and MWAIT disable**. Read-write. 1=The MONITOR and MWAIT opcodes become invalid. This affects what is reported back through CPUID Fn0000_0001_ECX[Monitor].
8	**IgnneEm: IGNNE port emulation enable**. Read-write. 1=Enable emulation of IGNNE port.
7:5	Reserved.
4	**INVD_WBINVD: INVD to WBINVD conversion**. Read-write. BIOS: See 2.3.3. 1=Convert INVD to WBINVD.
3	**TlbCacheDis: cacheable memory disable**. Read-write. 1=Disable performance improvement that assumes that the PML4, PDP, PDE and PTE entries are in cacheable memory. Operating systems that maintain page tables in uncacheable memory (UC memory type) must set the TlbCacheDis bit to insure proper operation.

2:1	Reserved.
0	**SmmLock: SMM code lock**. Read; write-1-only. SBIOS: 1. 1=SMI interrupts are not intercepted in SVM and the following SMM configuration registers are read-only: • MSRC001_0111 [SMM Base Address (SMM_BASE)] • MSRC001_0112 [SMM TSeg Base Address (SMMAddr)] • MSRC001_0113 [SMM TSeg Mask (SMMMask)] (all fields except MSRC001_0113[TClose:AClose]) • MSRC001_0116 [SMM Control (SMM_CTL)]

MSRC001_00[18,16] IO Range Registers Base (IORR_BASE[1:0])

Reset: 0000_0000_0000_0000h.

MSRC001_0016 and MSRC001_0017 combine to specify the first IORR range and MSRC001_0018 and MSRC001_0019 combine to specify the second IORR range. A CPU access--with address CPUAddr--is determined to be within IORR address range if the following equation is true:

CPUAddr[35:12] & PhyMask[35:12] == PhyBase[35:12] & PhyMask[35:12].

Bits	Description
63:36	RAZ.
35:12	**PhyBase[35:12]: physical base address**. Read-write.
11:5	RAZ.
4	**RdMem: read from memory**. Read-write. 1=Read accesses to the range are directed to system memory. 0=Read accesses to the range are directed to IO.
3	**WrMem: write to memory**. Read-write. 1=Write accesses to the range are directed to system memory. 0=Write accesses to the range are directed to IO.
2:0	RAZ.

MSRC001_00[19,17] IO Range Registers Mask (IORR_MASK[1:0])

Reset: 0000_0000_0000_0000h. See MSRC001_00[18,16].

Bits	Description
63:36	RAZ.
35:12	**PhyMask[35:12]: physical address mask**. Read-write.
11	**Valid**. Read-write. 1=The pair of registers that specifies an IORR range is valid.
10:0	RAZ.

MSRC001_001A Top Of Memory (TOP_MEM)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:36	RAZ.

35:23	**TOM[35:23]: top of memory**. Read-write. Specifies the address that divides between MMIO and DRAM. This value is normally placed below 4G. From TOM to 4G is MMIO; below TOM is DRAM. See 2.4.4 [System Address Map].
22:0	RAZ.

MSRC001_001D Top Of Memory 2 (TOM2)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:36	RAZ.
35:23	**TOM2[35:23]: second top of memory**. Read-write. Specifies the address divides between MMIO and DRAM. This value is normally placed above 4G. From 4G to TOM2 - 1 is DRAM; TOM2 and above is MMIO. See 2.4.4 [System Address Map]. This register is enabled by MSRC001_0010 [System Configuration (SYS_CFG)][MtrrTom2En].
22:0	RAZ.

MSRC001_001F Northbridge Configuration (NB_CFG)

Software is required to perform a read-modify-write in order to change any of the values in this register. Only one of these registers exists in multi-core devices; see 3.1.1 [Northbridge MSRs In Multi-Core Products].

Bits	Description
63:32	Alias of D18F3x8C.
31:0	Alias of D18F3x88.

MSRC001_0022 Machine Check Exception Redirection

Reset: 0000_0000_0000_0000h.

This register can be used to redirect machine check exceptions (MCEs) to SMIs or vectored interrupts. If both RedirSmiEn and RedirVecEn are set, then undefined behavior results.

Bits	Description
63:10	Reserved.
9	**RedirSmiEn**. Read-write. 1=Redirect MCEs (that are directed to this core) to generate an SMI-trigger IO cycle via MSRC001_0056. The status is stored in SMMFEC4[MceRedirSts].
8	**RedirVecEn**. Read-write. 1=Redirect MCEs (that are directed to this core) to generate a vectored interrupt, using the interrupt vector specified in RedirVector.
7:0	**RedirVector**. Read-write. See RedirVecEn.

MSRC001_00[35:30] Processor Name String

Reset: 0000_0000_0000_0000h.

These registers holds the CPUID name string in ASCII. The state of these registers are returned by CPUID instructions, CPUID Fn8000_000[4,3,2]_E[D,C,B,A]X. BIOS should set these registers to the product name

for the processor as provided by AMD. Each register contains a block of 8 ASCII characters; the least byte corresponds to the first ASCII character of the block; the most-significant byte corresponds to the last character of the block. MSRC001_0030 contains the first block of the name string; MSRC001_0035 contains the last block of the name string.

Bits	Description
63:0	**CpuNameString**. Read-write.

MSRC001_00[49:44] Machine Check Control Mask (MC*i*_CTL_MASK)

Table 116: Control register mapping for MSRC001_00[49:44]

Register	Function	MCi_CTL Register
MSRC001_0044	MC0_CTL_MASK	MSR0000_0400
MSRC001_0045	MC1_CTL_MASK	MSR0000_0404
MSRC001_0046	MC2_CTL_MASK	MSR0000_0408
MSRC001_0047	MC3_CTL_MASK	MSR0000_040C
MSRC001_0048	MC4_CTL_MASK	MSR0000_0410
MSRC001_0049	MC5_CTL_MASK	MSR0000_0414

These mask registers should be set up prior to enabling errors in MCi_CTL registers.

Bits	Description
63:32	**MSK[63:32]: Control Register Masks**. See: MSK[31:0]. Reset: 0000_0000h. BIOS: 0000_0000h.
31:0	**MSK[31:0]: Control Register Masks**. Reset: 0000_0000h. BIOS: 0000_0000h. Bits are read-only or read-write; corresponding to the attribute of the same bit in MCi_CTL. 1=Disable error logging in MCi_STATUS and MCi_ADDR for errors represented by the corresponding bit in MCi_CTL. This bit also disables error detection, and prevents error responses. See 2.16 [Machine Check Architecture]. For register layout, see the corresponding MCi_CTL register.

MSRC001_00[53:50] IO Trap (SMI_ON_IO_TRAP_[3:0])

Reset: 0000_0000_0000_0000h.

MSRC001_00[53:50] and MSRC001_0054 provide a mechanism for executing the SMI handler if a an access to one of the specified addresses is detected. Access address and access type checking is done before IO instruction execution. If the access address and access type match one of the specified IO address and access types, then: (1) the IO instruction is not executed; (2) any breakpoint, other than the single-step breakpoint, set on the IO instruction is not taken (the single-step breakpoint is taken after resuming from SMM); and (3) the SMI-trigger IO cycle specified byMSRC001_0056. The status is stored in SMMFEC4[IoTrapSts].

IO-space configuration accesses are special IO accesses. An IO access is defined as an IO-space configuration access when IO instruction address bits[31:0] are CFCh, CFDh, CFEh, or CFFh when IO-space configuration is enabled (IOCF8[ConfigEn]). The access address for a configuration space access is the current value of IOCF8[BusNo, Device, Function, RegNo]. The access address for an IO access that is not a configuration access is equivalent to the IO instruction address, bits[31:0].

The access address is compared with SmiAddr, and the instruction access type is compared with the enabled access types defined by ConfigSMI, SmiOnRdEn, and SmiOnWrEn. Access address bits[23:0] can be masked with SmiMask.

IO and configuration space trapping to SMI applies only to single IO instructions; it does not apply to string and REP IO instructions.

The conditional GP fault described by MSRC001_0015[IoCfgGpFault] takes priority over this trap.

Bits	Description
63	**SmiOnRdEn: enable SMI on IO read**. Read-write. 1=Enables SMI generation on a read access.
62	**SmiOnWrEn: enable SMI on IO write**. Read-write. 1=Enables SMI generation on a write access.
61	**ConfigSmi: configuration space SMI**. Read-write. 1=Configuration IO access. 0=Non-Configuration IO access.
60:56	Reserved.
55:32	**SmiMask[23:0]**. Read-write. SMI IO trap mask. 0=Mask address bit. 1=Do not mask address bit.
31:0	**SmiAddr[31:0]**. Read-write. SMI IO trap address.

MSRC001_0054 IO Trap Control (SMI_ON_IO_TRAP_CTL_STS)

Reset: 0000_0000_0000_0000h.

For each of the SmiEn bits below, 1=The trap specified by the corresponding MSR is enabled. See MSRC001_00[53:50].

Bits	Description
63:32	RAZ.
31:16	Reserved.
15	**IoTrapEn: IO trap enable**. Read-write. 1=Enable IO and configuration space trapping specified by MSRC001_00[53:50] and MSRC001_0054.
14:8	Reserved.
7	**SmiEn_3: SMI enable for the trap specified by MSRC001_0053**. Read-write.
6	Reserved.
5	**SmiEn_2: SMI enable for the trap specified by MSRC001_0052**. Read-write.
4	Reserved.
3	**SmiEn_1: SMI enable for the trap specified by MSRC001_0051**. Read-write.
2	Reserved.
1	**SmiEn_0: SMI enable for the trap specified by MSRC001_0050**. Read-write.
0	Reserved.

MSRC001_0055 Reserved

Bits	Description
63:0	Reserved.

MSRC001_0056 SMI Trigger IO Cycle

Reset: 0000_0000_0000_0000h.

See 2.4.6.2.3 [SMI Sources And Delivery]. This register specifies an IO cycle that may be generated when a local SMI trigger event occurs. If IoCycleEn is set and there is a local SMI trigger event, then the IO cycle generated is a byte read or write, based on IoRd, to address IoPortAddress. If the cycle is a write, then IoData contains the data written. If the cycle is a read, the value read is discarded. If IoCycleEn is clear and a local SMI trigger event occurs, then undefined behavior results.

Bits	Description
63:27	Reserved.
26	**IoRd: IO Read**. Read-write. 1=IO read; 0=IO write.
25	**IoCycleEn: IO cycle enable**. Read-write. 1=The SMI trigger IO cycle is enabled to be generated.
24	Reserved.
23:16	**IoData**. Read-write.
15:0	**IoPortAddress**. Read-write.

MSRC001_0058 MMIO Configuration Base Address

See 2.7 [Configuration Space] for a description of MMIO configuration space. All cores should program this register with the same value.

Bits	Description
63:36	RAZ.
35:20	**MmioCfgBaseAddr[35:20]: MMIO configuration base address bits[35:20]**. Read-write. Reset: 0. Specifies the base address of the MMIO configuration range. The size of the MMIO configuration-space address range is defined by MSRC001_0058[BusRange] as follows. All lower order undefined bits must be 0. BusRange / MmioCfgBaseAddr / BusRange / MmioCfgBaseAddr 0h / [35:20] / 5h / [35:25] 1h / [35:21] / 6h / [35:26] 2h / [35:22] / 7h / [35:27] 3h / [35:23] / 8h / [35:28] 4h / [35:24] / Fh-9h / Reserved
19:6	RAZ.
5:2	**BusRange: bus range identifier**. Read-write. Reset: 0. This specifies the number of buses in the MMIO configuration space range. The size of the MMIO configuration space range is 1 MB times the number of buses. Bits / Buses / Bits / Buses 0h / 1 / 5h / 32 1h / 2 / 6h / 64 2h / 4 / 7h / 128 3h / 8 / 8h / 256 4h / 16 / Fh-9h / Reserved
1	Reserved.
0	**Enable**. Read-write. Reset: 0. 1=MMIO configuration space is enabled.

MSRC001_0061 P-State Current Limit

See 2.5.3.1 [Core P-states].

Bits	Description
63:7	RAZ.
6:4	**PstateMaxVal: P-state maximum value**. Read; GP-write. Reset: Product-specific. Specifies the lowest-performance P-state (highest value) allowed. Attempts to change MSRC001_0062[PstateCmd] to a lower-performance P-state (higher value) are clipped to the value of this field.
3	RAZ.
2:0	**CurPstateLimit: current P-state limit**. Read; GP-write. Reset: 000b. Specifies the highest-performance P-state (lowest value) allowed. MSRC001_0061[CurPstateLimit] is always bounded by MSRC001_0061[PstateMaxVal]. Attempts to change the CurPstateLimit to a value greater (lower performance) than MSRC001_0061[PstateMaxVal] leaves CurPstateLimit unchanged. CurPstateLimit = D18F3x64[HtcPstateLimit] if D18F3x64[HtcAct]=1. The SB-TSI P-state limit register limits CurPstateLimit. See 2.10.2 [Sideband Temperature Sensor Interface (SB-TSI)].

MSRC001_0062 P-State Control

Bits	Description
63:3	MBZ.
2:0	**PstateCmd: P-state change command**. Read-write. Reset: Product-specific. Writes to this field cause the core to change to the indicated P-state number, specified by MSRC001_00[6B:64]. 0=SWP0, 1=SWP1, etc. P-state limits are applied to any P-state requests made through this register. See sections 2.5.3.1 [Core P-states]. Reads from this field return the last written value, regardless of whether any limits are applied.

MSRC001_0063 P-State Status

Bits	Description
63:3	RAZ.
2:0	**CurPstate: current P-state**. Read; GP-write; updated-by-hardware. Reset: values vary by product. Specifies the current P-state of the core. 0=P0, 1=P1, etc. When a P-state change is requested, the value in this field is updated once all required frequency and voltage transitions are completed. See 2.5.3.1 [Core P-states].

MSRC001_00[6B:64] P-State [7:0]

Reset: values vary by product.

All fields in these registers are required to be programmed to the same value in each core. See 2.5.3.1 [Core P-states].

Bits	Description
63	**PstateEn**. Read-write. 1=The P-state specified by this MSR is valid. 0=The P-state specified by this MSR is not valid. The purpose of this register is to indicate if the rest of the P-state information in the register is valid after a reset; it controls no hardware.
62:42	RAZ.
41:40	**IddDiv: current divisor field**. Read-write. After a reset, IddDiv and IddValue combine to specify the expected current dissipation of a single core that is in the P-state corresponding to the MSR number. These values are intended to be used to create ACPI-defined _PSS objects (see 2.5.3.1.7.2 [_PSS (Performance Supported States)]) and to perform the power delivery compatibility check (see 2.5.1.4 [Low Power Features]). The values are expressed in amps; they are not intended to convey final product power levels; they may not match the power levels specified in the Power and Thermal Datasheets. These fields, may be subsequently altered by software; they do not affect the hardware behavior. These fields are encoded as follows: Bits / Current Equation / Current Range 00b / IddValue / 1 A / 0 to 255 A 01b / IddValue / 10 A / 0 to 25.5 A 10b / IddValue / 100 A / 0 to 2.55 A 11b / Reserved
39:32	**IddValue: current value field**. Read-write. See IddDiv.
31:16	RAZ.
15:9	**CpuVid: CPU core VID**. Read-write. See 2.5.1 [Processor Power Planes And Voltage Control]. This field is required to be programmed as specified by MSRC001_0071[MaxVid, MinVid] otherwise undefined behavior results.
8:4	**CpuDidMSD: core divisor ID most significant digit**. Read-write. Specifies the integer part of the core clock divisor, see CpuDidLSD. Also see CpuDidLSD for special requirements associated with writing this field and for more details on the core clock divisor. Valid CpuDidMSD values are 19h-00h.

3:0	**CpuDidLSD: CPU core divisor ID least significant digit.** Read-write. Specifies the fractional part of the core clock divisor.
	• CpuDidLSD may only be programmed to 0000b, 0001b, 0010b, or 0011b. All other values are reserved.
	• The core clock divisor = CpuDidMSD + (CpuDidLSD * 0.25) + 1.
	  • For example, if CpuDidMSD = 00010 and CpuDidLSD = 0011, then the core clock divisor is 3.75.
	• Only the following core clock divisors can be created by CpuDidMSD and CpuDidLSD:
	  • 1.0-15.75 in 0.25 steps
	  • 16.0-26.5 in 0.50 steps (CpuDidLSD[0] has no effect in this range)
	• The CPU COF for this P-state can be calculated using the following equation:
	  • CPU COF = (main PLL frequency specified by D18F3xD4[MainPllOpFreqId]) / (core clock divisor specified by CpuDidMSD and CpuDidLSD).

MSRC001_0071 COFVID Status

See 2.5.3.1 [Core P-states].

Bits	Description
63:59	RAZ.

58:56	**CurPstateLimit: current P-state limit**. Read-only. Reset: 0. Provides the current lowest-performance P-state limit number. See MSRC001_0061[CurPstateLimit].
55	RAZ.
54:49	**MainPllOpFreqIdMax: main PLL operating frequency ID maximum**. Read-only. Reset: Product-specific. Specifies the maximum main PLL operating frequency supported by the processor. If MainPllOpFreqIdMax is greater than zero, the maximum frequency is 100 MHz * (MainPllOpFreqIdMax + 10h); if MainPllOpFreqIdMax = 00h, there is no frequency limit. See D18F3xD4[MainPllOpFreqId].
48:42	**MinVid: minimum voltage**. Read-only. Reset: Product-specific. Specifies the VID code corresponding to the minimum voltage (highest VID code) that the processor drives. 00h indicates that no minimum VID code is specified. See 2.5.1 [Processor Power Planes And Voltage Control].
41:35	**MaxVid: maximum voltage**. Read-only. Reset: Product-specific. Specifies the VID code corresponding to the maximum voltage (lowest VID code) that the processor drives. 00h indicates that no maximum VID code is specified. See 2.5.1 [Processor Power Planes And Voltage Control].
34:32	**StartupPstate: startup P-state number**. Read-only. Reset: Product-specific. Specifies the reset FID, DID, and VID for the core based on the P-state number selected. See MSRC001_00[6B:64].
31:25	**CurNbVid: current northbridge VID**. Read-only. Reset: Product-specific. Specifies the current VID code that has been sent to the voltage regulator for the VDDCR_NB plane.
24:21	Reserved.
20	**PstateInProgress**. Read-only. Reset: 0. Specifies whether a core voltage or frequency transition is in progress. 1=Change is in progress. 0=No changes in progress.
19	RAZ.
18:16	**CurPstate: current P-state**. Read-only. Reset: Product-specific. Specifies the current P-state requested by the core. See MSRC001_0063[CurPstate]. When a P-state change is requested, the value in this field is updated once all required frequency and voltage transitions are completed.
15:9	**CurCpuVid: current CPU core VID**. Read-only. Reset: Product-specific. Specifies the current VID code that has been sent to the voltage regulator. On a multi-core processor, CurCpuVid may not correspond to the VID code specified by the P-state currently requested by this core. When a P-state change is requested, the value in this field is updated after the voltage transition is complete, regardless of whether there is a following frequency transition.
8:4	**CurCpuDidMSD: current CPU core divisor ID most significant digit**. Read-only. Reset: Product-specific. Specifies the current CpuDidMSD of the core. See MSRC001_00[6B:64]. When a P-state change is requested, the value in this field is updated once all required frequency and voltage transitions are completed.
3:0	**CurCpuDidLSD: current CPU core divisor ID least significant digit**. Read-only. Reset: Product-specific. Specifies the current CpuDidLSD of the core. See MSRC001_00[6B:64]. When a P-state change is requested, the value in this field is updated once all required frequency and voltage transitions are completed.

MSRC001_0073 C-state Address

Reset: 0000_0000_0000_0000h.

Bits	Description
63:32	Reserved.
31:16	Reserved.

15:0	**CstateAddr: C-state address**. Read-write. BIOS: See 2.5.3.2.9. Specifies the IO addresses trapped by the core for C-state entry requests. A value of 0 in this field specifies that the core does not trap any IO addresses for C-state entry. Executing the HLT instruction still causes the core to enter a C-state. Writing values greater than 0FFF8h into this field results in undefined behavior. All other values cause the core to trap IO addresses CstateAddr through CstateAddr+7. See D18F4x118, D18F4x11C, and 2.5.3.2.2 [C-state Request Interface].

MSRC001_0074 CPU Watchdog Timer (CpuWdTmrCfg)

Reset: 0000_0000_0000_0000h.

The CPU watchdog timer (WDT) is implemented as a counter that counts out the time periods specified. The counter starts counting when CpuWdtEn is set. The counter does not count during halt or stop-grant. It restarts the count each time an operation of an instruction completes. If no operation completes by the specified time period, then a machine check error may be recorded if enabled (see MSR0000_0414 through MSR0000_0417). If a watchdog timer error overflow occurs (MSR0000_0415[Overflow]), a sync flood can be generated if enabled in D18F3x180[SyncFloodOnCpuLeakErr].

The CPU watchdog timer must be set higher than the NB watchdog timer (D18F3x44 [MCA NB Configuration]) in order to allow remote requests to complete. The CPU watchdog timer must be set the same for all CPUs in a system

Bits	Description
63:7	Reserved.
6:3	**CpuWdtCountSel: CPU watchdog timer count select**. Read-write. This, along with CpuWdtTimeBase, specifies the time period required for the WDT to expire. The time period is the value specified here times the time base specified by CpuWdtTimeBase. The actual timeout period may be anywhere from zero to one increments less than the values specified, due to non-deterministic behavior. The field is encoded as follows: Bits — Definition 0000b — 4095 0001b — 2047 0010b — 1023 0011b — 511 0100b — 255 0101b — 127 0110b — 63 0111b — 31 1000b — 8191 1001b — 16383 1111b-1010b — Reserved
2:1	**CpuWdtTimeBase: CPU watchdog timer time base**. Read-write. Specifies the time base for the timeout period specified in CpuWdtCountSel. Bits — Definition 00b — 1.31 ms 01b — 1.28 us 10b — 80 ns 11b — Reserved
0	**CpuWdtEn: CPU watchdog timer enable**. Read-write. 1=The WDT is enabled.

MSRC001_0111 SMM Base Address (SMM_BASE)

Reset: 0000_0000_0003_0000h.

This holds the base of the SMM memory region. The value of this register is stored in the save state on entry into SMM (see 2.4.6.2.5 [SMM Save State]) and it is restored on returning from SMM. The 16-bit CS (code segment) selector is loaded with SmmBase[19:4] on entering SMM. SmmBase[3:0] is required to be 0. The SMM base address can be changed in two ways:

- The SMM base address, at offset FF00h in the SMM state save area, may be changed by the SMI handler. The RSM instruction updates SmmBase with the new value.
- Normal WRMSR access to this register.

Bits	Description
63:32	Reserved.
31:0	**SmmBase**. IF (MSRC001_0015[SmmLock]==1) THEN Read-only ELSE Read-write. ENDIF.

MSRC001_0112 SMM TSeg Base Address (SMMAddr)

Reset: 0000_0000_0000_0000h.

See 2.4.6.2 [System Management Mode (SMM)] for information about SMM. See MSRC001_0113.

Each CPU access, directed at CPUAddr, is determined to be in the TSeg range if the following is true:

CPUAddr[35:17] & TSegMask[35:17] == TSegBase[35:17] & TSegMask[35:17].

For example, if TSeg spans 256K bytes and starts at the 1M byte address. The MSRC001_0112[TSegBase] would be set to 0010_0000h and the MSRC001_0113[TSegMask] to FFFC_0000h (with zeros filling in for bits[16:0]). This results in a TSeg range from 0010_0000 to 0013_FFFFh.

Bits	Description
63:36	Reserved.
35:17	**TSegBase[35:17]: TSeg address range base**. IF (MSRC001_0015[SmmLock]==1) THEN Read-only ELSE Read-write ENDIF.
16:0	Reserved.

MSRC001_0113 SMM TSeg Mask (SMMMask)

Reset: 0000_0000_0000_0000h.

See 2.4.6.2 [System Management Mode (SMM)] for information about SMM.

The ASeg address range is located at a fixed address from A0000h–BFFFFh. The TSeg range is located at a variable base (specified by MSRC001_0112[TSegBase]) with a variable size (specified by MSRC001_0113[TSegMask]). These ranges provide a safe location for SMM code and data that is not readily accessible by non-SMM applications. The SMI handler can be located in one of these two ranges, or it can be located outside these ranges. These ranges must never overlap each other.

This register specifies how accesses to the ASeg and TSeg address ranges are control as follows:

- If [A, T]Valid=0, then the address range is accessed as specified by MTRRs, regardless of whether the CPU is in SMM or not.
- If [A, T]Valid=1, then:
  - If in SMM, then:
    - If [A, T]Close=0, then the accesses are directed to DRAM with memory type as specified in [A, T]MTypeDram.
    - If [A, T]Close=1, then instruction accesses are directed to DRAM with memory type as specified in [A, T]MTypeDram and data accesses are directed at MMIO space and with attributes based on [A, T]MTypeIoWc.
  - If not in SMM, then the accesses are directed at MMIO space with attributes based on [A, T]MTypeIoWc.

Bits	Description
63:36	Reserved.
35:17	**TSegMask[35:17]: TSeg address range mask**. IF (MSRC001_0015[SmmLock]==1) THEN read-only ELSE Read-write ENDIF. See MSRC001_0112.
16:15	Reserved.
14:12	**TMTypeDram: TSeg address range memory type**. IF (MSRC001_0015[SmmLock]==1) THEN read-only ELSE Read-write ENDIF. Specifies the memory type for SMM accesses to the TSeg range that are directed to DRAM. The encoding is identical to the three LSBs of the MTRRs. See MSR0000_02[6F:68,59,58,50][2:0].
11	Reserved.
10:8	**AMTypeDram: ASeg Range Memory Type**. IF (MSRC001_0015[SmmLock]==1) THEN read-only ELSE Read-write ENDIF. Specifies the memory type for SMM accesses to the ASeg range that are directed to DRAM. The encoding is identical to the three LSBs of the MTRRs. See MSR0000_02[6F:68,59,58,50][2:0].
7:6	Reserved.
5	**TMTypeIoWc: non-SMM TSeg address range memory type**. IF (MSRC001_0015[SmmLock]==1) THEN read-only ELSE Read-write ENDIF. Specifies the attribute of TSeg accesses that are directed to MMIO space. 0=UC (uncacheable). 1=WC (write combining).
4	**AMTypeIoWc: non-SMM ASeg address range memory type**. IF (MSRC001_0015[SmmLock]==1) THEN read-only ELSE Read-write ENDIF. Specifies the attribute of ASeg accesses that are directed to MMIO space. 0=UC (uncacheable). 1=WC (write combining).
3	**TClose: send TSeg address range data accesses to MMIO**. Read-write. 1=When in SMM, direct data accesses in the TSeg address range to MMIO space. See AClose.
2	**AClose: send ASeg address range data accesses to MMIO**. Read-write. 1=When in SMM, direct data accesses in the ASeg address range to MMIO space. [A,T]Close allows the SMI handler to access the MMIO space located in the same address region as the [A, T]Seg. When the SMI handler is finished accessing the MMIO space, it must clear the bit. Failure to do so before resuming from SMM causes the CPU to erroneously read the save state from MMIO space.
1	**TValid: enable TSeg SMM address range**. IF (MSRC001_0015[SmmLock]==1) THEN read-only ELSE Read-write ENDIF. 1=The TSeg address range SMM enabled.
0	**AValid: enable ASeg SMM address range**. IF (MSRC001_0015[SmmLock]==1) THEN read-only ELSE Read-write ENDIF. 1=The ASeg address range SMM enabled.

MSRC001_0114 Virtual Machine Control (VM_CR)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:32	Reserved.
31:5	MBZ.
4	**Svme_Disable: SVME disable**. 1=MSRC000_0080[SVME] must be zero (MBZ) when writing to MSRC000_0080. Setting this bit when MSRC000_0080[SVME]=1 causes a #GP fault, regardless of the state of Lock. 0=MSRC000_0080[SVME] is read-write. See Lock.
3	**Lock: SVM lock**. Read-only; write-1-only; cleared-by-hardware. 1=Svme_Disable is read-only. 0=Svme_Disable is read-write.
2	**dis_a20m: disable A20 masking**. Read-write; set-by-hardware. 1=Disables A20 masking. This bit is set by hardware when the SKINIT instruction is executed.
1	**r_init: intercept INIT**. Read-write; set-by-hardware. This bit controls how INIT is delivered in host mode. This bit is set by hardware when the SKINIT instruction is executed. 0 = INIT delivered normally. 1 = INIT translated into a SX interrupt.
0	**DPD: debug port disable**. Read-write; set-by-hardware. This bit controls if debug facilities such as JTAG and HDT have access to the processor state information. This bit is set by hardware when the SKINIT instruction is executed. 0 = Debug port may be enabled. 1 = Debug port disabled; all mechanisms that could expose trusted code execution are disabled.

MSRC001_0115 IGNNE (IGNNE)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:32	Reserved.
31:1	MBZ.
0	**IGNNE: current IGNNE state**. Read-write. This bit controls the current state of the processor internal IGNNE signal.

MSRC001_0116 SMM Control (SMM_CTL)

IF (MSRC001_0015[SmmLock]==1) THEN GP-read-write. ELSE GP-Read; write-only. ENDIF.
The bits in this register are processed in the order of: smm_enter, smi_cycle, smm_dismiss, rsm_cycle and smm_exit. However, only the following combination of bits may be set in a single write (all other combinations result in undefined behavior):

- smm_enter and smi_cycle.
- smm_enter and smm_dismiss.
- smm_enter, smi_cycle and smm_dismiss.
- smm_exit and rsm_cycle.

Software is responsible for ensuring that smm_enter and smm_exit operations are properly matched and are not nested.

Bits	Description
63:5	MBZ.
4	**rsm_cycle: send RSM special cycle**. 1=Send a RSM special cycle.
3	**smm_exit: exit SMM**. 1=Exit SMM.
2	**smi_cycle: send SMI special cycle**. 1=Send a SMI special cycle.
1	**smm_enter: enter SMM**. 1=Enter SMM.
0	**smm_dismiss: clear SMI**. 1=Clear the SMI pending flag.

MSRC001_0117 Virtual Machine Host Save Physical Address (VM_HSAVE_PA)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:36	MBZ.
35:12	**VM_HSAVE_PA: physical address of host save area**. Read-write. This register contains the physical address of a 4-KB page where VMRUN saves host state and where vmexit restores host state from. Writing this register causes a #GP if any of the lower 12 bits are not zero or if the address written is greater than F_FFFF_F000h.
11:0	MBZ.

MSRC001_0118 SVM Lock Key

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**SvmLockKey: SVM lock key**. RAZ; write. Writes to this register when MSRC001_0114[Lock]=0 write the SvmLockKey. Writes to this register when MSRC001_0114[Lock]=1 and SvmLockKey!=0 cause hardware to clear MSRC001_0114[Lock] if the value written is the same as the value stored in SvmLockKey.

MSRC001_011A Local SMI Status

Reset: 0000_0000_0000_0000h. This registers returns the same information that is returned in SMMFEC4 [Local SMI Status] portion of the SMM save state. The information in this register is only updated when MSRC001_0116[smm_dismiss] is set by software.

Bits	Description
63:32	RAZ.
31:0	See: SMMFEC4 [Local SMI Status].

MSRC001_0140 OS Visible Work-around MSR0 (OSVW_ID_Length)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:16	Reserved.
15:0	**OSVW_ID_Length: OS visible work-around ID length**. Read-write. See the *Revision Guide for AMD Family 14h Models 00h-0Fh Processors* for the definition of this field.

MSRC001_0141 OS Visible Work-around MSR1 (OSVW Status)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**OsvwStatusBits: OS visible work-around status bits**. Read-write. See the *Revision Guide for AMD Family 14h Models 00h-0Fh Processors* for the definition of this field.

3.23 MSRs - MSRC001_1xxx

MSRC001_1004 CPUID Features (Features)

MSRC001_1004 and MSRC001_1005 provide some control over values read from CPUID functions.

Bits	Description
63:32	**FeaturesEcx**. Read-write. Provides back-door control over the features reported in CPUID function 0000_0001_ECX (see CPUID Fn0000_0001_ECX).
31:0	**FeaturesEdx**. Read-write. Provides back-door control over the features reported in CPUID function 0000_0001_EDX (see CPUID Fn0000_0001_EDX).

MSRC001_1005 Extended CPUID Features (ExtFeatures)

See MSRC001_1004.

Bits	Description
63:32	**ExtFeaturesEcx**. Read-write. Provides back-door control over the features reported in CPUID function 8000_0001_ECX (see CPUID Fn8000_0001_ECX).
31:0	**ExtFeaturesEdx**. Read-write. Provides back-door control over the features reported in CPUID function 8000_0001_EDX (see CPUID Fn8000_0001_EDX).

MSRC001_1020 Load-Store Configuration

Reset: 0000_0000_0000_0000h.

Bits	Description
63:29	Reserved.
28	**DisStreamSt**. Read-write. BIOS: See 2.3.3. 1=Streaming store functionality disabled.
27:0	Reserved.

MSRC001_1021 Instruction Cache Configuration

Reset: 0000_0000_1000_8000h.

Bits	Description
63:10	Reserved.
9	**DIS_SPEC_TLB_RLD**. Read-write. BIOS: See 2.3.3. Disable speculative TLB reloads.
8:0	Reserved.

MSRC001_1022 Data Cache Configuration

Reset: 0600_0018_0000_0000h.

Bits	Description
63:14	Reserved.
13	**DIS_HW_PF**. Read-write. BIOS: See 2.3.3. 1=Disable hardware prefetches.
12:0	Reserved.

MSRC001_1030 IBS Fetch Control (IbsFetchCtl)

Reset: 0000_0000_0000_0000h.

The IBS fetch sampling engine selects an instruction fetch to profile when the engine's periodic fetch counter reaches IbsFetchMaxCnt. The periodic fetch counter is an internal 20 bit counter that increments after every fetch cycle that completes when IbsFetchEn=1 and IbsFetchVal=0. When the selected instruction fetch completes or is aborted, the status of the fetch is written to the IBS fetch registers (this register, MSRC001_1031 and MSRC001_1032) and an interrupt is generated. The interrupt service routine associated with this interrupt is responsible for saving the performance information stored in IBS fetch registers. See 2.6.2 [Instruction Based Sampling (IBS)]

Bits	Description
63:58	Reserved.
57	**IbsRandEn: random instruction fetch tagging enable**. Read-write. 1=Bits 3:0 of the fetch counter are randomized when IbsFetchEn is set to start the fetch counter. 0=Bits 3:0 of the fetch counter are set to 0h when IbsFetchEn is set to start the fetch counter.
56	**IbsL2TlbMiss: instruction cache L2TLB miss**. Read-write. 1=The instruction fetch missed in the L2 TLB.
55	**IbsL1TlbMiss: instruction cache L1TLB miss**. Read-write. 1=The instruction fetch missed in the L1 TLB.
54:53	**IbsL1TlbPgSz: instruction cache L1TLB page size**. Read-write. This field indicates the page size of the translation in the L1 TLB. This field is only valid if IbsPhyAddrValid=1. Bits — Definition 00b — 4 KB 01b — 2 MB 1Xb — Reserved
52	**IbsPhyAddrValid: instruction fetch physical address valid**. Read-write. 1=The physical address in MSRC001_1032 and the IbsL1TlbPgSz field are valid for the instruction fetch.

51	**IbsIcMiss: instruction cache miss**. Read-write. 1=The instruction fetch missed in the instruction cache.
50	**IbsFetchComp: instruction fetch complete**. Read-write. 1=The instruction fetch completed and the data is available for use by the instruction decoder.
49	**IbsFetchVal: instruction fetch valid**. Read-write; set-by-hardware. Reset: 0. 1=New instruction fetch data available. When this bit is set, the fetch counter stops counting and an interrupt is generated as specified bythe APIC LVT specified by MSRC001_103A[LvtOffset]. This bit must be cleared and IbsFetchCnt must be written to 0000h for the fetch counter to start counting again.
48	**IbsFetchEn: instruction fetch enable**. Read-write. 1=Instruction fetch sampling is enabled.
47:32	**IbsFetchLat: instruction fetch latency**. Read-write. This field indicates the number of clock cycles from when the instruction fetch was initiated to when the data was delivered to the core. If the instruction fetch is abandoned before the fetch completes, this field returns the number of clock cycles from when the instruction fetch was initiated to when the fetch was abandoned.
31:16	**IbsFetchCnt**. Read-write; updated-by-hardware. This field returns the current value of bits 19:4 of the periodic fetch counter.
15:0	**IbsFetchMaxCnt**. Read-write. This field specifies maximum count value of the periodic fetch counter. Programming this field to 0000h and setting IbsFetchEn results in undefined behavior. Bits 19:4 of the maximum count are programmed in the field. Bits 3:0 of the maximum countare always 0000.

MSRC001_1031 IBS Fetch Linear Address (IbsFetchLinAd)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:12	**IbsFetchLinAd[63:12]: instruction fetch linear address**. Read-write. See IbsFetchLinAd[11:0].
11:0	**IbsFetchLinAd[11:0]: instruction fetch linear address**. Read-write. This field provides the linear address in canonical form for the tagged instruction fetch.

MSRC001_1032 IBS Fetch Physical Address (IbsFetchPhysAd)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:36	Reserved.
35:12	**IbsFetchPhysAd[35:12]: instruction fetch physical address**. Read-write. This provides the physical address for the tagged instruction fetch. This field contains valid data only if MSRC001_1030[IbsPhyAddrValid] is asserted.
11:0	**IbsFetchPhysAd[11:0]**. Read-only. Alias of MSRC001_1031[IbsFetchLinAd[11:0]]. Lower physical address is not modified by address translation, so they are always the same as the linear address.

MSRC001_1033 IBS Execution Control (IbsOpCtl)

Reset: 0000_0000_0000_0000h. The IBS execution sampling engine tags a micro-op that is issued in the next cycle to profile when the engine's periodic op counter reaches IbsOpMaxCnt. The periodic op counter is an internal 27 bit counter that increments every cycle when IbsOpEn=1 and IbsOpVal=0 and rolls over when the counter reaches IbsOpMaxCnt. When the periodic op counter rolls over bits 6:0 of the counter are randomized

by hardware. When the micro-op is retired, the status of the operation is written to the IBS execution registers (this register, MSRC001_1034, MSRC001_1035, MSRC001_1036, MSRC001_1037, MSRC001_1038 and MSRC001_1039) and an interrupt is generated as specified by MSRC001_103A. The interrupt service routine associated with this interrupt is responsible for saving the performance information stored in IBS execution registers. See 2.6.2 [Instruction Based Sampling (IBS)].

Bits	Description
63:59	Reserved.
58:52	**IbsOpCurCntExt: periodic op counter current count extension**. Read-write. This field returns the current value of bits 26:20 of the periodic op counter.
51:32	**IbsOpCurCnt: periodic op counter current count**. Read-write. Reset: X. This field returns the current value of the periodic op counter.
31:27	Reserved.
26:20	**IbsOpMaxCntExt: periodic op counter maximum count extension**. Read-write. This field specifies maximum count value of the periodic op counter. Bits 26:20 of the maximum count are programmed in the field.
19	**IbsOpCntCtl: periodic op counter count control**. Read-write. Reset: 0b. 1=Count dispatched ops 0=Count clock cycles.
18	**IbsOpVal: micro-op sample valid**. Read-write; set-by-hardware. 1=New instruction execution data available. When this bit is set, the periodic op counter stops counting until software clears the bit and an interrupt is generated as specified bythe APIC LVT specified by MSRC001_103A[LvtOffset].
17	**IbsOpEn: micro-op sampling enable**. Read-write. 1=Instruction execution sampling enabled.
16	Reserved.
15:0	**IbsOpMaxCnt: periodic op counter maximum count**. Read-write. Reset: X. This field specifies maximum count value of the periodic op counter. Bits 19:4 of the maximum count are programmed in the field. Bits 3:0 of the maximum count are always 0000.

MSRC001_1034 IBS Op Logical Address (IbsOpRip)

Reset: xxxx_xxxx_xxxx_xxxxh.

Bits	Description
63:0	**IbsOpRip: micro-op linear address**. Read-write. Linear address in canonical form for the instruction that contains the tagged micro-op.

MSRC001_1035 IBS Op Data (IbsOpData)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:39	Reserved.
38	**IbsRipInvalid: RIP invalid**. Read-write. 1=MSRC001_1034[IbsOpRip] is not valid for the tagged micro-op.
37	**IbsOpBrnRet: branch micro-op retired**. Read-write. 1=Tagged operation was a branch micro-op that retired.

36	**IbsOpBrnMisp: mispredicted branch micro-op**. Read-write. 1=Tagged operation was a branch micro-op that was mispredicted.
35	**IbsOpBrnTaken: taken branch micro-op**. Read-write. 1=Tagged operation was a branch micro-op that was taken.
34	**IbsOpReturn: return micro-op**. Read-write. 1=Tagged operation was return micro-op.
33	**IbsOpMispReturn: mispredicted return micro-op**. Read-write. 1=Tagged operation was a mispredicted return micro-op.
32	**IbsOpBrnResync: resync micro-op**. Read-write. 1=Tagged operation was resync micro-op.
31:16	**IbsTagToRetCtr: micro-op tag to retire count**. Read-write. This field returns the number of cycles from when the micro-op was tagged to when the micro-op was retired. This field is equal to IbsCompToRetCtr when the tagged micro-op is a NOP.
15:0	**IbsCompToRetCtr: micro-op completion to retire count**. Read-write. This field returns the number of cycles from when the micro-op was completed to when the micro-op was retired.

MSRC001_1036 IBS Op Data 2 (IbsOpData2)

Reset: 0000_0000_0000_0000h.

Northbridge data is only valid for load operations that miss both the L1 data cache and the L2 cache. If a load operation crosses a cache line boundary, the data returned in this register is the data for the access to the lower cache line.

Bits	Description
63:32	Reserved.
31:3	Reserved.
2:0	**NbIbsReqSrc: Northbridge IBS request data source**. Read-write. Bits — Definition 0h — No valid status 1h — Reserved 2h — Cache: data returned from a CPU cache 3h — DRAM: data returned from DRAM 4h — Reserved 5h — Reserved 6h — Reserved 7h — Other: data returned from MMIO/Config/PCI/APIC

MSRC001_1037 IBS Op Data 3 (IbsOpData3)

Reset: 0000_0000_0000_0000h.
If a load or store operation crosses a 128-bit boundary, the data returned in this register is the data for the access to the data below the 128-bit boundary.

Bits	Description
63:48	Reserved.
47:32	**IbsDcMissLat: data cache miss latency**. Read-write. This field indicates the number of clock cycles from when a miss is detected in the data cache to when the data was delivered to the core. The value returned by this counter is not valid for data cache writes.

31:19	Reserved.
18	**IbsDcPhyAddrValid: data cache physical address valid**. Read-write. 1=The physical address in MSRC001_1039 is valid for the load or store operation.
17	**IbsDcLinAddrValid: data cache linear address valid**. Read-write. 1=The linear address in MSRC001_1038 is valid for the load or store operation.
16	**IbsDcMabHit: MAB hit**. Read-write. 1=The tagged load or store operation hit on an already allocated MAB. IBS data in MSRC001_1036 is not valid if this bit is set.
15	**IbsDcLockedOp: locked operation**. Read-write. 1=Tagged load or store operation is a locked operation.
14	**IbsDcUcMemAcc: UC memory access**. Read-write. 1=Tagged load or store operation accessed uncacheable memory.
13	**IbsDcWcMemAcc: WC memory access**. Read-write. 1=Tagged load or store operation accessed write combining memory.
12	Reserved.
11	**IbsDcStToLdFwd: data forwarded from store to load operation**. Read-write. Reset: X. 1=Data for tagged load operation was forwarded from a store operation. If this bit is set and IbsDcStToLdCan=1, then the data for the load operation forwarded from a store operation but the data was not forwarded immediately.
10:9	Reserved.
8	**IbsDcMisAcc: misaligned access**. Read-write. 1=The tagged load or store operation crosses a 128 bit address boundary.
7	**IbsDcMiss: data cache miss**. Read-write. 1=The cache line used by the tagged load or store was not present in the data cache.
6	**IbsDcL2tlbHit2M: data cache L2TLB hit in 2M page**. Read-write. 1=The physical address for the tagged load or store operation was present in a 2M page table entry in the data cache L2TLB.
5	Reserved.
4	**IbsDcL1tlbHit2M: data cache L1TLB hit in 2M page**. Read-write. 1=The physical address for the tagged load or store operation was present in a 2M page table entry in the data cache L1TLB.
3	**IbsDcL2tlbMiss: data cache L2TLB miss**. Read-write. 1=The physical address for the tagged load or store operation was not present in the data cache L2TLB.
2	**IbsDcL1tlbMiss: data cache L1TLB miss**. Read-write. 1=The physical address for the tagged load or store operation was not present in the data cache L1TLB.
1	**IbsStOp: store op**. Read-write. 1=Tagged operation is a store operation
0	**IbsLdOp: load op**. Read-write. 1=Tagged operation is a load operation

MSRC001_1038 IBS DC Linear Address (IbsDcLinAd)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**IbsDcLinAd**. Read-write. This field provides the linear address in canonical form for the tagged load or store operation. This field contains valid data only if MSRC001_1037[IbsDcLinAddrValid] is asserted.

MSRC001_1039 IBS DC Physical Address (IbsDcPhysAd)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:36	Reserved.
35:0	**IbsDcPhysAd: load or store physical address**. Read-write. Provides the physical address for the tagged load or store operation. The lower 12 bits are not modified by address translation, so they are always the same as the linear address. This field contains valid data only if MSRC001_1037[IbsDcPhyAddrValid] is asserted.

MSRC001_103A IBS Control

Reset: 0000_0000_0000_0000h. Read; GP-write. This register returns the value of D18F3x1CC.

Bits	Description
63:9	Reserved.
8	**LvtOffsetVal: local vector table offset valid**. 1=The offset in LvtOffset is valid.
7:4	Reserved.
3:0	**LvtOffset: local vector table offset**. This specifies the address of the IBS LVT entry in the APIC registers as follows: Bits Definition 3h-0h LVT address = <500h + LvtOffset<<4> Fh-4h Reserved See APIC[530:500]).

MSRC001_103B IBS Branch Target Address

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**IbsBrTarget**. Read-write. This provides the logical address in canonical form for the branch target. This field contains a valid branch target address when the tagged operation is a branch and the field is non-zero.

3.24 Performance Counter Events

3.24.1 CPU Performance Counter Events

This section provides the performance counter events that may be selected through MSRC001_00[03:00] [Performance Event Select (PERF_CTL[3:0])][EventSelect and UnitMask]. See MSRC001_00[03:00] and MSRC001_00[07:04] [Performance Event Counter (PERF_CTR[3:0])].

3.24.1.1 Floating Point Events

See the following events for additional floating point information:
- PMCx0CB [Retired a Floating Point Instruction].
- PMCx0DB [FPU Exceptions].

PMCx000 Dispatched FPU Operations

The number of operations (uops) dispatched to the FPU execution pipelines. This event reflects how busy the FPU pipelines are. This includes all operations done by x87, MMX™ and SSE instructions, including moves. Each increment represents a one-cycle dispatch event; packed 128-bit SSE operations count as two ops in 64-bit FPU implementations; scalar operations count as one. This event is a speculative event (see PMCx0CB). Since this event includes non-numeric operations it is not suitable for measuring MFLOPs.

UnitMask	Description
7:2	Reserved.
1	Pipe1 (fmul, store, mmx) ops.
0	Pipe0 (fadd, imul, mmx) ops.

PMCx001 Cycles in which the FPU is Empty

The number of cycles in which the FPU is empty. Invert this (MSRC001_00[03:00][Invert]=1) to count cycles in which at least one FPU operation is present in the FPU.

PMCx002 Dispatched Fast Flag FPU Operations

The number of FPU operations that use the fast flag interface (e.g. FCOMI, COMISS, COMISD, UCOMISS, UCOMISD, MOVD, CVTSD2SI). This event is a speculative event.

PMCx003 Retired SSE Operations

The number of SSE operations retired. This counter can count either FLOPS (UnitMask[6] = 1) or uops (UnitMask[6] = 0).

UnitMask	Description
7	Reserved.
6	Op type: 0=uops. 1=FLOPS
5	Double precision divide/square root ops.
4	Double precision multiply ops.
3	Double precision add/subtract ops.
2	Single precision divide/square root ops.
1	Single precision multiply ops.
0	Single precision add/subtract ops.

PMCx004 Retired Move Ops

The number of move uops retired.

UnitMask	Description
7:4	Reserved.
3	All other move uops
2	All other merging move uops

1	Reserved.
0	Reserved.

PMCx005 Retired Serializing Ops

The number of serializing uops retired. A bottom-executing uop is not issued until it is the oldest non-retired uop in the FPU. A control-renaming uop requires a rename from a limited pool of control renames.

UnitMask	Description
7:4	Reserved.
3	x87 control-renaming uops retired.
2	x87 bottom-executing uops retired.
1	SSE control-renaming uops retired.
0	SSE bottom-executing uops retired.

PMCx011 Retired x87 Floating Point Operations

The number of x87 floating point ops that have retired.

UnitMask	Description
7:3	Reserved.
2	Divide and fsqrt ops.
1	Multiply ops.
0	Add/subtract ops.

3.24.1.2 Load/Store Events

See 3.24.1.4 [System Interface Events] for the following:

- PMCx065 [Memory Requests by Type].
- PMCx068 [MAB Requests].

PMCx020 Segment Register Loads

The number of segment register loads performed.

UnitMask	Description
7	Reserved.
6	HS
5	GS
4	FS
3	DS
2	SS
1	CS
0	ES

PMCx021 Pipeline Restart Due to Self-Modifying Code

The number of pipeline restarts that were caused by self-modifying code (a store that hits any instruction that's been fetched for execution beyond the instruction doing the store).

PMCx022 Pipeline Restart Due to Probe Hit

The number of pipeline restarts caused by invalidating probes that hit load out-of-order with respect to other load.

PMCx023 RSQ Full

The number of cycles that the RSQ holds retired stores. This buffer holds stores waiting to retire as well as requests that missed the data cache and are waiting on a refill. This condition does not stall further data cache accesses, but does stall retirement of stores.

PMCx024 Locked Operations

This event covers locked operations performed and their execution time. The execution time represented by the cycle counts is typically overlapped to a large extent with other instructions. The non-speculative cycles event is suitable for event-based profiling of lock operations that tend to miss in the cache.

UnitMask	Description
7:3	Reserved.
2	The number of cycles to unlock $line (not including cache miss)
1	The number cycles to acquire bus lock
0	The number of locked instructions executed

PMCx026 Retired CLFLUSH Instructions

The number of CLFLUSH instructions retired.

PMCx027 Retired CPUID Instructions

The number of CPUID instructions retired.

PMCx02A Store to Load Forward Operations block Loads

Counts the number store to load forward operations block loads since data was available.

UnitMask	Description
7:3	Reserved.
2	Misaligned.
1	Store is smaller than load.
0	Address mismatches (starting byte not the same).

3.24.1.3 Data Cache Events

PMCx040 Data Cache Accesses

The number of accesses to the data cache for load and store references. This may include certain microcode scratchpad accesses, although these are generally rare. Each increment represents an eight-byte access, although the instruction may only be accessing a portion of that. This event is a speculative event.

PMCx041 Data Cache Misses

The number of data cache references which miss in the data cache and allocate a MAB. This event is a speculative event.

PMCx042 Data Cache Refills from L2 or Northbridge

The number of data cache refills satisfied from the L2 cache (and/or the northbridge), per the UnitMask. The UnitMask selects lines in one or more specific coherency states. Each increment reflects a 64-byte transfer. If UnitMask[0] is selected it might be less than 64-bytes. This event is a speculative event.

UnitMask	Description
7:5	Reserved.
4	Modified
3	Owned
2	Exclusive
1	Shared
0	Non-cacheable return of data.

PMCx043 Data Cache Refills from the northbridge

The number of L1 cache refills satisfied from the northbridge (DRAM or another processor's cache), as opposed to the L2. The UnitMask selects lines in one or more specific coherency states. Each increment reflects a 64-byte transfer. This event is a speculative event.

UnitMask	Description
7:5	Reserved.
4	Modified
3	Owned
2	Exclusive
1	Shared
0	Non-cacheable read data.

PMCx044 Data Cache Lines Evicted

The UnitMask may be used to count only victims in specific coherency states. Each increment represents a 64-byte transfer. Lines brought into the data cache by PrefetchNTA instructions are evicted directly to system

memory (if dirty) or invalidated (if clean). This event is a speculative event.

UnitMask	Description
7:5	Reserved.
4	Modified eviction
3	Owned eviction
2	Exclusive eviction
1	Shared eviction
0	Evicted from probe

PMCx045 L1 DTLB Miss and L2 DTLB Hit

The number of data cache accesses that miss in the L1 DTLB and hit in the L2 DTLB. This event is a speculative event.

PMCx046 DTLB Miss

The number of data cache accesses that miss in both the L1 and L2 DTLBs. This event is a speculative event.

UnitMask	Description
7:4	Reserved.
3	Count loads that miss L2TLB.
2	Count stores that miss L2TLB.
1	Count loads that miss L1TLB.
0	Count stores that miss L1TLB.

PMCx047 Misaligned Accesses

The number of data cache accesses that are misaligned. These are accesses which cross a sixteen-byte boundary. They incur an extra cache access (reflected in PMCx040), and an extra cycle of latency on reads. This event is a speculative event.

PMCx04B Prefetch Instructions Dispatched

The number of prefetch instructions dispatched by the decoder. Such instructions may or may not cause a cache line transfer. All Dcache and L2 accesses, hits and misses by prefetch instructions, except for prefetch instructions that collide with an outstanding hardware prefetch, are included in these events. This event is a speculative event.

UnitMask	Description
7:3	Reserved.
2	NTA (PrefetchNTA)
1	Store (PrefetchW)
0	Load (Prefetch, PrefetchT0/T1/T2)

PMCx04C DCACHE Misses by Locked Instructions

The number of cacheable locked operations that miss in the data cache. Cacheable locks are defined as locks that hit write back memory space and are not misaligned.

PMCx04D L1 DTLB Hit

The number of data cache accesses that hit in the L1 DTLB. This event is a speculative event.

UnitMask	Description
7:2	Reserved.
1	L1 2M TLB hit
0	L1 4K TLB hit

PMCx052 DCACHE Ineffective Software Prefetchs

The number of software prefetches that do not cause an actual data cache refill. The unit mask may be used to determine the specific cause.

UnitMask	Description
7:4	Reserved.
3	SW Prefetch hit in L2.
2	SW prefetches that don't get a MAB and don't cause PMCx052[1,0].
1	Software prefetch hit a pending fill.
0	Software prefetch hit in the data cache.

PMCx054 Global Page Invalidations

This event counts TLB flushes that flush TLB entries that have the global bit set.

3.24.1.4 System Interface Events

PMCx065 Memory Requests by Type

These events reflect accesses to memory of each region type (as defined by MTRR or PAT settings). Unit-Mask[1] and UnitMask[7] reflect full 64 byte writes or full or partial 32 byte writes.

UnitMask	Description
7	Streaming store (SS) requests.
6:2	Reserved.
1	Request to write-combining (WC) memory.
0	Requests to non-cacheable (UC) memory.

PMCx068 MAB Requests

The PMCx068 and PMCx069 events provide a measure of the average L1 refill latency for instruction cache (IC) and data cache (DC) misses. The UnitMask is an encoded value which selects one of the ten Miss Address Buffers (MABs) that handle L1 cache misses. PMCx068 counts the number of misses handled by the selected MAB; PMCx069 counts the number of processor cycles the selected MAB is busy waiting for the refill response.

When used together, these two events provide a measure of the average refill latency seen by the selected MAB. Dividing the cycle count by the number of misses measured over the same interval with the same Unit-Mask gives the average refill latency in processor clock cycles. The refill times include cases that are satisfied from higher cache levels as well as from system DRAM. Measurements using UnitMask 00h and 08h give the most direct indication of average system latency for DC and IC cache refills, respectively. Measurements using other MABs typically include queuing delays caused by resource contention with prior refills.

It is not meaningful to combine the UnitMask values. Use UnitMask 0Ah or 0Bh to measure combined events from all DC or IC buffers, respectively.

UnitMask	Description
7:0	Buffer N, where N = 00h to 0Bh. Bits — Definition 07h-00h — DC miss buffers 0 to 7 09h-08h — IC miss buffers 0 to 1 0Ah — Any DC miss buffer 0Bh — Any IC miss buffer FFh-0Ch — Reserved

PMCx069 MAB Wait Cycles

See PMCx068.

PMCx06C System Response by Coherence State

The number of responses from the northbridge for cache refill requests. The UnitMask may be used to select specific cache coherency states. Each of {Exclusive, Modified, Shared} represents one 64-byte cache line transferred from the northbridge (DRAM or from the cache of another core) to the data cache or instruction cache.

- Modified responses may be for Dcache store miss refills, PrefetchW software prefetches, or Change-to-Dirty requests that get a dirty probe hit in another cache.
- Exclusive responses may be for any Icache refill, Dcache load miss refill, other software prefetches.
- Shared responses may be for any request that hits a clean line in another cache.
- Change-to-Dirty success response is for a Dcache upgrade request (store hit to a shared line).
- Uncacheable response is for all uncacheable system requests.

UnitMask	Description
7	Reserved.
6	Uncacheable.
5	Change-to-Dirty success.
4	Data Error.

3	Owned.
2	Shared.
1	Modified.
0	Exclusive.

PMCx076 CPU Clocks not Halted

The number of clocks that the CPU is not in a halted state (due to STPCLK or a HLT instruction). Note: this event allows system idle time to be automatically factored out from IPC (or CPI) measurements, providing the OS halts the CPU when going idle. If the OS goes into an idle loop rather than halting, such calculations are influenced by the IPC of the idle loop.

PMCx07D Requests to L2 Cache

The number of requests to the L2 cache for Icache or Dcache fills.

UnitMask	Description
7:4	Reserved.
3	Tag snoop request.
2	Reserved.
1	DC fill
0	IC fill

PMCx07E L2 Cache Misses

The number of requests that miss in the L2 cache.

UnitMask	Description
7:2	Reserved.
1	DC fill.
0	IC fill.

PMCx07F L2 Fill/Writeback

The number of lines written into the L2 cache due to victim writebacks from the Icache or Dcache, or write-backs of dirty lines from the L2 to the system (UnitMask[1]). Each increment represents a 64-byte cache line transfer.

UnitMask	Description
7:4	Reserved.
3	IC attribute writes which store into the L2. Different from PMCx07F UnitMask[2] as the line must be valid and not dirty in the L2 for the attributes to be written.
2	IC attribute writes which access the L2. Different than IC event to count evictions (PMCx08B) as evictions can be dropped by the BU.
1	L2 Writebacks to system.

0	L2 fills (victims from L1 caches).

PMCx162 PDC Miss

Counts the number of PDC miss events specified by the UnitMask.

UnitMask	Description
7	Reserved.
6	Guest: PML4E Level.
5	Guest: PDPE Level.
4	Guest: PDE Level.
3	Reserved.
2	Host: PML4E Level.
1	Host: PDPE Level.
0	Host: PDE Level.

3.24.1.5 Instruction Cache Events

Note: All instruction cache events are speculative events unless specified otherwise.

PMCx080 Instruction Cache Fetches

The number of successful instruction cache accesses by the instruction fetcher that result in data being sent to the decoder. Each access is an aligned 32 byte read, from which a varying number of instructions may be decoded.

PMCx081 Instruction Cache Misses

The number of instruction fetches and prefetch requests that miss in the instruction cache. This is typically equal to or very close to the sum of events 82h and 83h. Each miss results in a 64-byte cache line refill.

PMCx082 Instruction Cache Refills from L2

The number of instruction cache refills satisfied from the L2 cache. Each increment represents one 64-byte cache line transfer.

PMCx083 Instruction Cache Refills from System

The number of instruction cache refills from system memory (or another cache).
Each increment represents one 64-byte cache line transfer.

PMCx085 ITLB Miss

The number of instruction fetches that miss in the 4K ITLB and 2M ITLB.

UnitMask	Description

7:2	Reserved.
1	Instruction fetches to a 2M page.
0	Instruction fetches to a 4K page.

PMCx087 Instruction Fetch Stall

The number of cycles the instruction fetcher is stalled. This may be for a variety of reasons such as branch predictor updates, unconditional branch bubbles, far jumps and cache misses, among others. May be overlapped by instruction dispatch stalls or instruction execution, such that these stalls don't necessarily impact performance.

PMCx088 Return Stack Hits

The number of near return instructions (RET or RET Iw) that get their return address from the return address stack (i.e. where the stack has not gone empty). This may include cases where the address is incorrect (return mispredicts). This may also include speculatively executed false-path returns. Return mispredicts are typically caused by the return address stack underflowing, however they may also be caused by an imbalance in calls vs. returns, such as doing a call but then popping the return address off the stack.

Note: This event cannot be reliably compared with events C9h and CAh (such as to calculate percentage of return mispredicts due to an empty return address stack), since it may include speculatively executed false-path returns that are not included in those retire-time events.

PMCx089 Return Stack Overflows

The number of (near) call instructions that cause the return address stack to overflow. When this happens, the oldest entry is discarded. This count may include speculatively executed calls.

PMCx08B Instruction Cache Victims

The number of cachelines evicted from the instruction cache to the L2.

PMCx08C Instruction Cache Lines Invalidated

The number of instruction cache lines invalidated.

UnitMask	Description
7:2	Reserved.
1	IC invalidate due to a BU probe.
0	IC invalidate due to an LS probe.

PMCx099 ITLB Reloads

The number of ITLB reload requests.

PMCx09A ITLB Reloads Aborted

The number of ITLB reloads aborted.

3.24.1.6 Execution Unit Events

See 3.24.1.2 [Load/Store Events] for the following:
- PMCx026 [Retired CLFLUSH Instructions].
- PMCx027 [Retired CPUID Instructions].

See 3.24.1.4 [System Interface Events] for the following:
- PMCx076 [CPU Clocks not Halted].

PMCx0C0 Retired Instructions

The number of instructions retired (execution completed and architectural state updated). This count includes exceptions and interrupts - each exception or interrupt is counted as one instruction.

PMCx0C1 Retired uops

The number of micro-ops retired. This includes all processor activity (instructions, exceptions, interrupts, microcode assists, etc.).

PMCx0C2 Retired Branch Instructions

The number of branch instructions retired. This includes all types of architectural control flow changes, including exceptions and interrupts.

PMCx0C3 Retired Mispredicted Branch Instructions

The number of branch instructions retired, of any type, that were not correctly predicted in either target or direction. This includes those for which prediction is not attempted (far control transfers, exceptions and interrupts), and excludes resyncs.

PMCx0C4 Retired Taken Branch Instructions

The number of taken branches that were retired. This includes all types of architectural control flow changes, including exceptions and interrupts, and excludes resyncs.

PMCx0C5 Retired Taken Branch Instructions Mispredicted

The number of retired taken branch instructions that were mispredicted, and excludes resyncs.

PMCx0C6 Retired Far Control Transfers

The number of far control transfers retired including far call/jump/return, IRET, SYSCALL and SYSRET, plus exceptions and interrupts, and excludes resyncs. Far control transfers are not subject to branch prediction.

PMCx0C7 Retired Branch Resyncs

The number of resync branches. These reflect pipeline restarts due to certain microcode assists and events such as writes to the active instruction stream, among other things. Each occurrence reflects a restart penalty similar to a branch mispredict. This is relatively rare.

PMCx0C8 Retired Near Returns

The number of near return instructions (RET or RET Iw) retired.

PMCx0C9 Retired Near Returns Mispredicted

A near return instruction was retired that mispredicted in either target or direction.

PMCx0CA Retired Mispredicted Taken Branch Instructions due to Target Mismatch

A taken branch instruction was retired that mispredicted in target address (but not in direction).

PMCx0CB Retired a Floating Point Instruction

A floating point (x87, MMX, or SSE) instruction was retired. The UnitMask allows the distinction between x87/MMX and SSE instructions.

Since this event includes non-numeric instructions it is not suitable for measuring MFLOPS.

UnitMask	Description
7:2	Reserved.
1	SSE floating point instruction was retired (SSE, SSE2, SSE3, MNI)
0	x87 or MMX™ instruction was retired

PMCx0CD Interrupts-Masked Cycles

The number of processor cycles where interrupts are masked (EFLAGS.IF = 0). Using edge-counting with this event gives the number of times IF is cleared; dividing the cycle-count value by this value gives the average length of time that interrupts are disabled on each instance. Compare the edge count with PMCx0CF to determine how often interrupts are disabled for interrupt handling vs. other reasons (e.g. critical sections).

PMCx0CE Interrupts-Masked Cycles with Interrupt Pending

The number of processor cycles where interrupts are masked (EFLAGS.IF = 0) and an interrupt is pending. Using edge-counting with this event and comparing the resulting count with the edge count for PMCx0CD gives the proportion of interrupts for which handling is delayed due to prior interrupts being serviced, critical sections, etc. The cycle count value gives the total amount of time for such delays. The cycle count divided by the edge count gives the average length of each such delay.

PMCx0CF Interrupts Taken

The number of hardware interrupts taken. This does not include software interrupts (INT n instruction).

PMCx0DB FPU Exceptions

The number of floating point unit exceptions for microcode assists. The UnitMask may be used to isolate specific types of exceptions.

UnitMask	Description
7:4	Reserved.
3	SSE and x87 microtraps
2	SSE reclass microfaults
1	SSE retype microfaults
0	x87 reclass microfaults

PMCx0DC DR0 Breakpoint Matches

The number of matches on the address in breakpoint register DR0, per the breakpoint type specified in DR7. The breakpoint does not have to be enabled. Each instruction breakpoint match incurs an overhead of about 120 cycles; load/store breakpoint matches do not incur any overhead.

PMCx0DD DR1 Breakpoint Matches

The number of matches on the address in breakpoint register DR1. See notes for PMCx0DC.

PMCx0DE DR2 Breakpoint Matches

The number of matches on the address in breakpoint register DR2. See notes for PMCx0DC.

PMCx0DF DR3 Breakpoint Matches

The number of matches on the address in breakpoint register DR3. See notes for PMCx0DC.

3.24.2 NB Performance Counter Events

3.24.2.1 Memory Controller Events

If more than one UnitMask bit is set for an event, then simultaneous events are counted only once.

PMCx0E0 DRAM Accesses

The number of memory accesses performed by the local DRAM controller. The UnitMask may be used to isolate the different DRAM page access cases. Page miss cases incur an extra latency to open a page; page conflict cases incur both a page-close as well as page-open penalties. These penalties may be overlapped by DRAM accesses for other requests and don't necessarily represent lost DRAM bandwidth. The associated penalties are as follows:

Page miss: Trcd (DRAM RAS-to-CAS delay)
Page conflict: Trp + Trcd (DRAM row-precharge time plus RAS-to-CAS delay)

UnitMask	Description
7	Read request.

6	Write request.
5:3	Reserved.
2	DCT0 Page Conflict.
1	DCT0 Page Miss.
0	DCT0 Page hit.

PMCx0E1 DRAM Controller Page Table Events

The number of page table events in the local DRAM controller. This table maintains information about which DRAM pages are open. An overflow occurs when a request for a new page arrives when the page table becomes full, as the oldest entry is speculatively closed. Each occurrence reflects an access latency penalty equivalent to a page conflict.

UnitMask	Description
7:5	Reserved.
4	DCT0 Page table is closed due to row inactivity.
3	DCT0 Page table idle cycle limit decremented.
2	DCT0 Page table idle cycle limit incremented.
1	DCT0 Number of stale table entry hits. (hit on a page closed too soon).
0	DCT0 Page Table Overflow.

PMCx0E2 Memory Controller DRAM Command Slots Missed

UnitMask	Description
7	Reserved.
6	DCT0 Prefetch.
5	Reserved.
4	DCT0 RBD.
3:0	Reserved.

PMCx0E3 Memory Controller Turnarounds

The number of turnarounds on the local DRAM data bus. The UnitMask may be used to isolate the different cases. These represent lost DRAM bandwidth, which may be calculated as follows (in bytes per occurrence):

DIMM turnaround: DRAM_width_in_bytes * 2 edges_per_memclk * 2
R/W turnaround: DRAM_width_in_bytes * 2 edges_per_memclk * 1
R/W turnaround: DRAM_width_in_bytes * 2 edges_per_memclk * (Tcl-1)

where DRAM_width_in_bytes is 8 or 16 (for single- or dual-channel systems), and Tcl is the CAS latency of the DRAM in memory system clock cycles (where the memory clock for DDR-400, or PC3200 DIMMS, for example, would be 200 MHz).

UnitMask	Description
7:2	Reserved.

1	DCT0 write-to-read turnaround.
0	DCT0 read-to-write turnaround.

PMCx0E4 Memory Controller RBD Queue Events

UnitMask	Description
7:4	Reserved.
3	Bank is closed due to bank conflict with an outstanding request in the RBD queue.
2	D18F2x94[DcqBypassMax] counter reached.
1:0	Reserved.

PMCx0E8 Thermal Status

UnitMask	Description
7	PROCHOT_L asserted by an external source and the assertion causes a P-state change.
6	Number of clocks HTC P-state is active.
5	Number of clocks HTC P-state is inactive.
4:3	Reserved.
2	Number of times the HTC transitions from inactive to active.
1	Reserved.
0	MEMHOT_L assertions.

PMCx0E9 CPU/IO Requests to Memory/IO

These events reflect request flow between units, as selected by the UnitMask. It is not possible to tell from these events how much data is going in which direction, as there is no distinction between reads and writes. Also, particularly for IO, the requests may be for varying amounts of data, anywhere from one to sixty-four bytes.

UnitMask	Description
7:4	Reserved.
3	CPU to Mem.
2	CPU to IO.
1	IO to Mem.
0	IO to IO.

PMCx0EA Cache Block Commands

The number of requests made to the system for cache line transfers or coherency state changes, by request type. Each increment represents one cache line transfer, except for Change-to-Dirty. If a Change-to-Dirty request hits on a line in another processor's cache that's in the Owned state, it causes a cache line transfer, otherwise there is no data transfer associated with Change-to-Dirty requests.

UnitMask	Description
7:6	Reserved.
5	Change to Dirty (first store to clean block already in cache).
4	Read Block Modified (Dcache store miss refill).
3	Read Block Shared (Icache refill).
2	Read Block (Dcache load miss refill).
1	Reserved.
0	Victim Block (Writeback).

PMCx0EB Sized Commands

The number of Sized Read/Write commands handled by the System Request Interface (local processor and hostbridge interface to the system). These commands may originate from the processor or hostbridge. See PMCx0EC, which provides a separate measure of Hostbridge accesses.

UnitMask	Description
7:6	Reserved.
5	SzRd DW (1-16 doublewords). Block-oriented DMA reads, typically cache line size.
4	SzRd Byte (4 bytes). Legacy or mapped IO.
3	Posted SzWr DW (1-16 doublewords). Block-oriented DMA writes, often cache line sized; also processor Write Combining buffer flushes.
2	Posted SzWr Byte (1-32 bytes). Sub-cache-line DMA writes, size varies; also flushes of partially-filled Write Combining buffer.
1	Non-Posted SzWr DW (1-16 doublewords). Legacy or mapped IO, typically 1 doubleword.
0	Non-Posted SzWr Byte (1-32 bytes). Legacy or mapped IO, typically 1-4 bytes.

PMCx0EC Probe Responses and Upstream Requests

This covers two unrelated sets of events: cache probe results, and requests received by the hostbridge from devices on a non-coherent link.

Probe results: These events reflect the results of probes sent from a memory controller to local caches. They provide an indication of the degree data and code is shared between processors (or moved between processors due to process migration). The dirty-hit events indicate the transfer of a 64-byte cache line to the requester (for a read or cache refill) or the target memory (for a write). The system bandwidth used by these, in terms of bytes per unit of time, may be calculated as 64 times the event count, divided by the elapsed time. Sized writes to memory that cover a full cache line do not incur this cache line transfer -- they simply invalidate the line and are reported as clean hits. Cache line transfers occur for Change2Dirty requests that hit cache lines in the Owned state. (Such cache lines are counted as Modified-state refills for PMCx06C, System Read Responses.)

Upstream requests: The upstream read and write events reflect requests originating from a device on a local link. DMA accesses may be anywhere from 1 to 64 bytes in size, but may be dominated by a particular size such as 32 or 64 bytes, depending on the nature of the devices.

UnitMask	Description
7	Upstream low priority writes.
6	Reserved.
5	Upstream low priority reads.
4	Upstream high priority reads.
3	Probe hit dirty with memory cancel (probed by DMA read or cache refill request).
2	Probe hit dirty without memory cancel (probed by Sized Write or Change2Dirty).
1	Probe hit clean.
0	Probe miss.

PMCx0EE DEV Events

UnitMask	Description
7	Reserved.
6	DEV error
5	DEV miss
4	DEV hit
3:0	Reserved.

PMCx1F0 Memory Controller Requests

Sized Read/Write activity: The Sized Read/Write events reflect 32- or 64-byte transfers (as opposed to other sizes which could be anywhere between 1 and 64 bytes), from either the processor or the Hostbridge. Such accesses from the processor would be due only to write combining buffer flushes, where 32-byte accesses would reflect flushes of partially-filled buffers. Event 65h provides a count of sized write requests associated with WC buffer flushes; comparing that with counts for these events (providing there is very little Hostbridge activity at the same time) give an indication of how efficiently the write combining buffers are being used. Event 65h may also be useful in factoring out WC flushes when comparing these events with the Upstream Requests component of event ECh.

UnitMask	Description
7	Reserved.
6	64 Byte Sized Reads
5	32 Bytes Sized Reads
4	64 Bytes Sized Writes
3	32 Bytes Sized Writes
2:0	Reserved.

3.24.2.2 Crossbar Events

PMCx1E9 Sideband Signals and Special Cycles

UnitMask	Description
7:5	Reserved.
4	INVD
3	WBINVD
2	SHUTDOWN
1	STOPGRANT
0	Reserved.

PMCx1EA Interrupt Events

UnitMask	Description
7	EOI
6	INT
5	STARTUP
4	INIT
3	NMI
2	SMI
1	LPA
0	Fixed and LPA

4 Register List

The following is a list of all storage elements, context, and registers provided in this document. Page numbers, register mnemonics, and register names are provided.