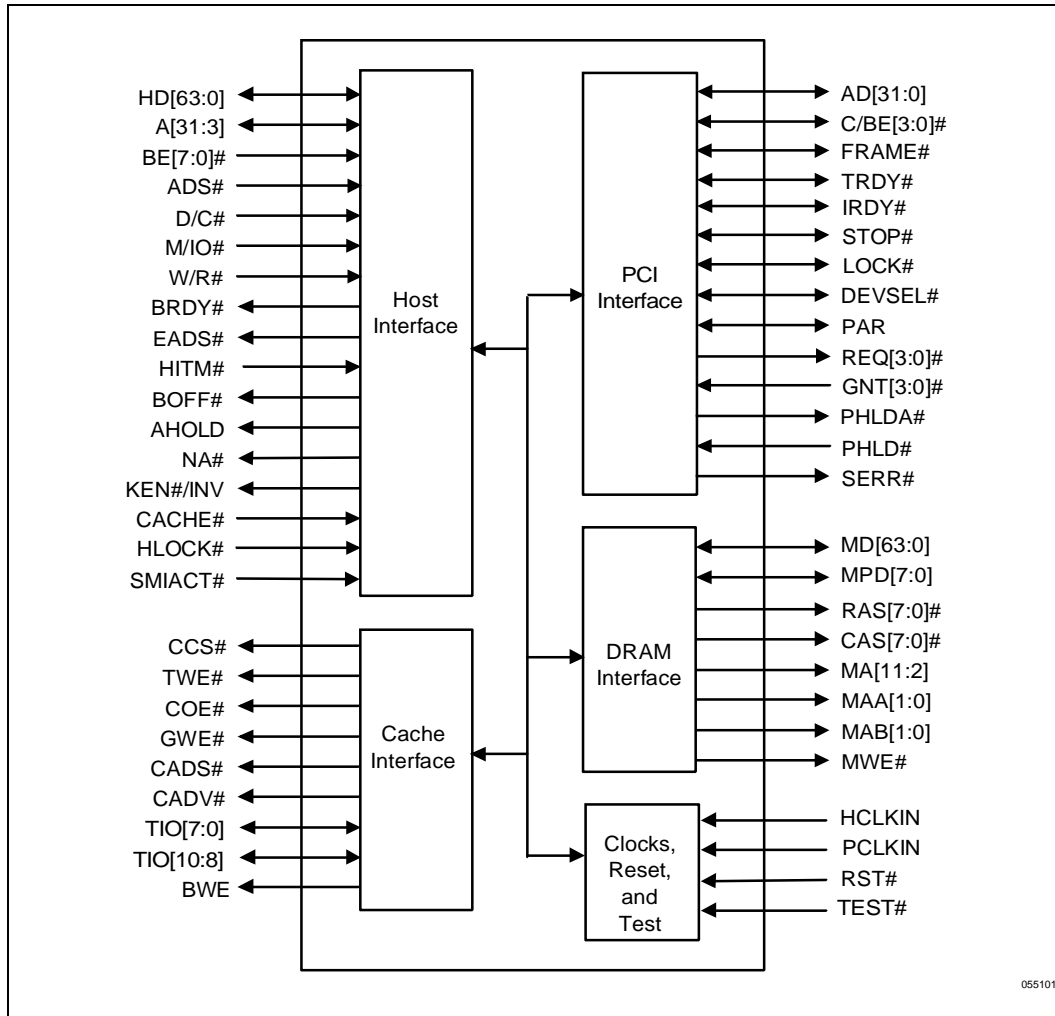# intel®

# INTEL 430HX PCISET
# 82439HX SYSTEM CONTROLLER (TXC)

- **Supports All 3V Pentium® Processors**
- **Dual Processor Support**
- **PCI 2.1 Compliant**
- **Integrated Second-Level Cache Controller**
  - **Direct Mapped Organization**
  - **Write-Back Cache Policy**
  - **Cacheless, 256 KB, and 512 KB**
  - **Pipelined Burst SRAMs**
  - **Cache Hit Read/Write Cycle Timings at 3-1-1-1**
  - **Back-to-Back Read Cycles at 3-1-1-1-1-1-1-1**
  - **Integrated Tag/Valid Status Bits for Cost Savings and Performance**
  - **Optional 512-MB DRAM Cacheability Limit**
  - **Supports 5V SRAMs for Tag Address**
- **Integrated DRAM controller**
  - **4-MB to 512-MB Main Memory**
  - **64-Mb DRAM Technology Support**
  - **8-QWord Deep Merging DRAM Write Buffer**
  - **Enhanced EDO/Hyper Page Mode DRAM; 4-2-2-2 Reads and x-2-2-2 Writes at 60 MHz; 5-2-2-2 Reads and x-2-2-2 Writes at 66 MHz**
  - **8 RAS Lines**
  - **Integrated Programmable-Strength Memory Address Buffers**
  - **CAS-Before-RAS Refresh**
- **Optional Parity**
- **Single 324-Pin BGA Package**

- **Optional Error Checking and Correction (ECC)**
  - **Superior DRAM Data Integrity**
  - **Single Bit Error Correction, Multi-Bit Error Detection plus Nibble Failure Detection ECC Code**
  - **Single and Multi-Bit Error Reporting**
  - **Virtual Swapable Bank Support (i.e., can swap out problem banks)**
  - **Merging Write Buffer Eliminates Most Partial Writes Cycles**
- **Fully Synchronous, Minimum Latency 25/30/33 MHz PCI Bus Interface**
  - **Zero Wait State CPU-to-PCI Write Timings (no IRDY stall) for Superior Graphics Performance**
  - **Enhanced CPU-to-PCI Read Latencies for Superior Graphics/PIO Performance**
  - **21-DWord PCI-DRAM Post Buffer**
  - **22-DWord PCI-to-DRAM Read Prefetch Buffer**
  - **Write-Back Merging for PCI to DRAM Writes**
  - **Write-Back Forwarding for PCI to DRAM Reads**
  - **Pipelined Snoop Ahead**
  - **Multi-Transaction Timer to Support Multiple Short PCI Transactions Within the Same PCI Arbitration Cycle**
- **Supports the Universal Serial Bus (USB)**
- **Supported Kits**
  - **82439HX ISA Kit (TXC, PIIX3)**
  - **82439HX ISA/DP Kit (TXC, PIIX3, IOAPIC)**

The Intel 430HX PCIset consists of the 82439HX System Controller (TXC) and the 82371SB PCI I/O IDE Xcelerator (PIIX3). The TXC is a single-chip host-to-PCI bridge and provides the second level cache control and DRAM control functions. The second level (L2) cache controller supports a write-back cache policy for cache sizes of 256 Kbytes and 512 Kbytes. Cacheless designs are also supported. The cache memory is implemented with synchronous pipelined burst SRAMs. An external Tag RAM is used for the address tag and an internal Tag RAM for the cache line status bits. The TXC provides a 64/72-bit data path to main memory and memory sizes up to 512 Mbytes. The DRAM controller provides eight rows and optional DRAM Error detection/correction or parity. The TXC's optimized PCI interface allows the CPU to sustain the highest possible bandwidth to the graphics frame buffer at all frequencies. Using the snoop ahead feature, The TXC allows PCI masters to achieve full PCI bandwidth. For increased system performance, the TXC contains read prefetch and posted write buffers.

intel®

**82439HX TXC Simplified Block Diagram**

# CONTENTS

## REVISION HISTORY

| Revision Date | Version | Description |
|---|---|---|
| July 1996 | -001 | Initial Release |
| April 1997 | -002 | Included information from Specification Update |

The Intel 430HX PCIset device may contain design defects or errors known as errata. Characterized errata that may cause the 430HX's behavior to deviate from published specifications are documented in the "Intel 430HX PCIset: 82439HX (TXC) Specification Update" (Order number 297652).

## 1.0. ARCHITECTURE OVERVIEW

The TXC interfaces with the Pentium® processor host bus, a dedicated memory data bus, and the PCI bus (Figure 1). The TXC connects directly to the Pentium processor 3V host bus, directly to 5V or 3V DRAMs, and directly to the 5V PCI bus. The Intel 430HX PCIset consists of the 82439HX TXC and the PCI IDE/ISA Xcellerator (PIIX3) components. PIIX3 provides the PCI-to-ISA bridge functions along with other features such as a fast IDE interface, Plug-n-Play port, APIC interface, Universal Serial Bus (USB) and PCI 2.1 Compliance operation.

**Data Flow**

Processor cycles are sent directly to the second level cache with control for the second level cache provided by the TXC. All other processor cycles are sent to their destination (DRAM, PCI or internal TXC configuration space) via the TXC. PCI Master cycles are sent to main memory through the TXC. The TXC performs snoop or inquire cycles using the host bus.
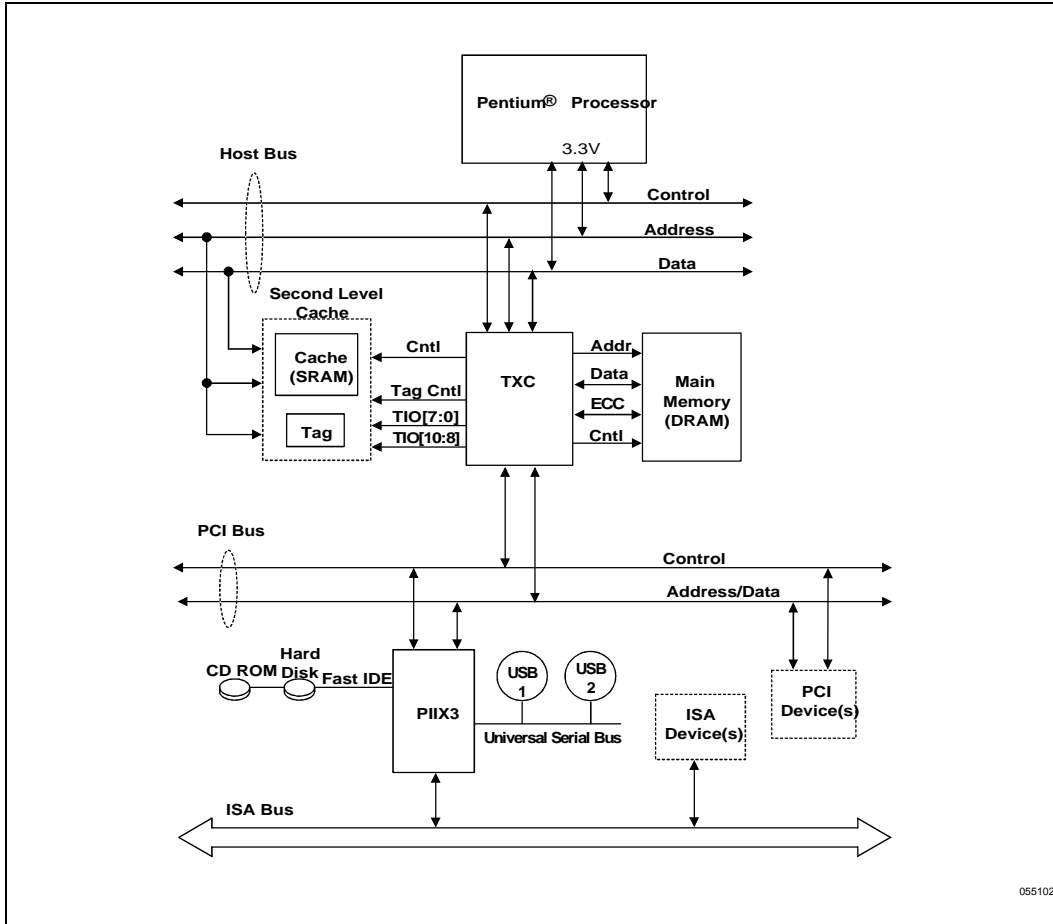


**Figure 1. TXC System Block Diagram**

**DRAM Interface**

The DRAM interface is a 64/72-bit data path that supports both standard page mode and Extended Data Out (EDO) memory. The DRAM interface supports 4 Mbytes to 512 Mbytes with 8 RAS lines and also supports symmetrical and asymmetrical addressing for 1M, 2M, and 4M deep SIMMs and symmetrical addressing for 16-Mbyte deep SIMMs.

**Second Level Cache**

The TXC supports a write-back cache policy providing all necessary snoop functions and inquire cycles. The second level cache is direct mapped and supports both a 256-Kbyte or 512-Kbyte SRAM configuration using pipelined burst SRAMs. The burst 256-Kbyte configuration performance is 3-1-1-1 for read/write cycles; pipelined back-to-back reads can maintain a 3-1-1-1-1-1-1-1 transfer rate. An optional mode extends the DRAM L2 cacheability range to 512 Mbytes.

**PCI Interface**

The PCI interface is 2.1 compliant and supports up to 4 PCI bus masters in addition to the PIIX3 bus master requests. The PCI-to-DRAM interface can reach a 112 Mbyte/sec transfer rate for reads and 121 Mbytes/sec for writes.

**Data Path and Buffers**

The TXC data path is optimized for minimum latency and maximum throughput operation from both the CPU and PCI masters. The TXC contains two physical sets of buffers for optimizing data flow. A 6-DWord buffer is provided for CPU-to-PCI writes that helps maximize the graphic writes to PCI bandwidth. An 8-QWord deep *merging* memory buffer is provided that is used for CPU-to-main memory writes, write-back cycles (Posted at 3111), PCI-to-main memory write posting, and PCI-from-main memory read prefetching.

**Error Detection and Correction**

Parity or error correction are software configurable options (parity is the default). The ECC mode provides single-error correction, double-error detection, and detection of all errors confined to a single nibble for the DRAM memory subsystem.

## 2.0. SIGNAL DESCRIPTION

This section provides a detailed description of each signal. The signals are arranged in functional groups according to their associated interface.

The "#" symbol at the end of a signal name indicates that the active, or asserted state occurs when the signal is at a low voltage level. When "#" is not present after the signal name, the signal is asserted when at the high voltage level.

The terms assertion and negation are used extensively. This is done to avoid confusion when working with a mixture of "active-low" and "active-high" signals. The term **assert**, or **assertion** indicates that a signal is active, independent of whether that level is represented by a high or low voltage. The term **negate**, or **negation** indicates that a signal is inactive.

All 3V output signals can drive 5V TTL inputs.

## 2.1. Host Interface

| Name | Type | Description |
|------|------|-------------|
| A[31:3] | I/O 3V TTL | **Address Bus:** A[31:3] connects to the address bus of the CPU. During CPU cycles A[31:3] are inputs. The TXC drives A[31:3] during inquire cycles on behalf of PCI initiators. Strapping options on A[31:28] are described in the Cache Control Register Description section. Strapping options on A27 are described in the DRAMC Register Description. |
| BE[7:0]# | I 3V TTL | **Byte Enables:** The CPU byte enables indicate which byte lane the current CPU cycle is accessing. All eight byte lanes must be provided to the CPU if the cycle is a cacheable read regardless of the state of BE[7:0]#. |
| ADS# | I 3V TTL | **Address Status:** The CPU asserts ADS# in T1 of the CPU bus cycle. |
| BRDY# | O 3V TTL | **Bus Ready:** The TXC asserts BRDY# to indicate to the CPU that data is available on reads or has been received on writes. |
| NA# | O 3V TTL | **Next Address:** The TXC asserts NA# to the CPU to enable pipelining. |
| AHOLD | O 3V TTL | **Address Hold:** The TXC asserts AHOLD when a PCI initiator is performing a cycle to DRAM. |
| EADS# | O 3V TTL | **External Address Strobe:** Asserted by the TXC to inquire the first level cache when servicing PCI master references of DRAM. |
| BOFF# | O 3V TTL | **Back Off:** Asserted by the TXC when required to terminate a CPU cycle that was in progress. |
| HITM# | I 3V TTL | **Hit Modified:** Asserted by the CPU to indicate that the address presented with the last assertion of EADS# is modified in the first level cache and needs to be written back. |

| Name | Type | Description |
|------|------|-------------|
| M/IO#, D/C#, W/R# | I 3V TTL | **Memory/IO; Data/Control; Write/Read:** Asserted by the CPU with ADS# to indicate the type of cycle that the system needs to perform. |
| HLOCK# | I 3V TTL | **Host Lock:** All CPU cycles sampled with the assertion of HLOCK# and ADS#, until the negation of HLOCK# must be atomic (i.e., no PCI activity to DRAM is allowed). |
| CACHE# | I 3V TTL | **Cacheable:** CACHE# is asserted by the CPU during a read cycle to indicate the CPU will perform a burst line fill. Asserted by the CPU during a write cycle to indicate the CPU will perform a burst write-back cycle. |
| KEN#/INV | O 3V TTL | **Cache Enable:** KEN#/INV functions as both the KEN# signal during CPU read cycles, and the INV signal during L1 snoop cycles. During CPU cycles, KEN#/INV is normally low. KEN#/INV is driven high(low) during the EADS# assertion of a PCI master DRAM write (read) snoop cycle. |
| SMIACT# | I 3V TTL | **System Management Interrupt Active:** The CPU asserts SMIACT# when it is in system management mode as a result of an SMI. |
| HD[63:0] | I/O 3V TTL | **Host Data:** These signals are connected to the CPU data bus. |

**NOTES:**
All of the signals in the host interface are described in the Pentium processor datasheet. The preceding table highlights TXC specific uses of these signals.

## 2.2. DRAM Interface

| Name | Type | Description |
|------|------|-------------|
| MPD[7:0] | I/O 3V/5V TTL | **MEMORY PARITY DATA:** These signals connect to the DRAM parity or ECC bits. The MPD pins have similar timing and drive capability to the MD pins. ECC versus parity operation is determined by PCON[DDIM] field. These signals have internal weak pulldowns, the same as the existing MD signals. These signals are always driven during write cycles, regardless of the existence of parity memory. |
| RAS[7:0]# | O 3V | **Row Address Strobe:** These pins select the DRAM row. |
| CAS[7:0]# | O 3V | **Column Address Strobe:** These pins select the DRAM column. |
| MA[11:2] | O 3V | **Memory Address:** This is the row and column address for DRAM. These buffers include programmable size selection. |
| MAA[1:0] | O 3V | **Memory Address Copy A:** One copy of the MAs that change during a burst read or write of DRAM. These buffers include programmable size selection. |
| MAB[1:0] | O 3V | **Memory Address Copy B.** A second copy of the MAs that change during a burst read or write of DRAM. These buffers include programmable size selection. |

| Name | Type | Description |
|------|------|-------------|
| MWE# | O<br>3V | **Memory Write Enable.** MWE# should be used as the write enable for the memory data bus. This signal has a programmable size selection (see DRAMEC[MAD] field). |
| MD[63:0] | I/O<br>3V/5V<br>TTL | **Memory Data:** These signals are connected to the DRAM data bus. |

## 2.3. Secondary Cache Interface

| Name | Type | Description |
|------|------|-------------|
| CADV# | O<br>3V | **Cache Advance:** Assertion causes the PBSRAM in the secondary cache to advance to the next QWord in the cache line. |
| CADS# | O<br>3V | **Cache Address Strobe:** Assertion causes the PBSRAM in the secondary cache to load the PBSRAM address register from the PBSRAM address pins. |
| CCS# | O<br>3V | **Cache Chip Select:** When this signal is asserted and CADS# is asserted, the second level cache will power up (if necessary). When this signal is negated and CADS# is asserted, the second level cache will power down. When CCS# is negated the second level cache ignores ADS#. If CCS# is asserted when ADS# is asserted the second level cache will power up (if necessary) and perform an access. |
| COE# | O<br>3V | **Cache Output Enable:** The secondary cache data RAMs drive the CPUs data bus when COE# is asserted. |
| GWE# | O<br>3V | **Global Write Enable:** GWE# assertion causes all the byte lanes to be written into the secondary cache data RAMs independent of BE[7:0]#. |
| BWE# | O<br>3V | **Byte Write Enable:** Asserted low with GWE#=high to enable using host's BE[7:0]# to control byte lanes to pipeline burst SRAM cache. |
| TIO[10:8] | I/O<br>3V5V<br>TTL | **Extended Tag Address:** These are inputs during CPU accesses and outputs during second level cache line fills and second level cache line invalidates due to inquire cycles. TIO[10:8] contain the optional extension tag address to increase cacheability limit to 512 Mbytes. TIO[10:8] is enabled via the ECE bit of the CC register. When the extended tag address is not enabled, these bits are not used, but are tied internally low to prevent them from floating. TIO[9:8] have internal pull-down resistors. TIO[10] requires an external pull-down resistor for the extended cacheability and a pull-up resistor for the normal cacheability/DRAM cache. The extended cacheability and DRAM cache functions are mutually exclusive. |
| TIO[7:0] | I/O<br>3V/5V<br>TTL | **Tag Address:** These are inputs during CPU accesses and outputs during second level cache line fills and second level cache line invalidates due to inquire cycles. TIO[7:0] have internal pull-down resistors. |
| TWE# | O<br>5V<br>TTL | **Tag Write Enable:** When asserted, new state and tag addresses are written into the external tag. |

## 2.4. PCI Interface

| Name | Type | Description |
|---|---|---|
| AD[31:0] | I/O 5V | **Address/Data:** The standard PCI address and data lines. Address is driven with FRAME# assertion, data is driven or received in following clocks. |
| C/BE[3:0]# | I/O 5V | **Command/Byte Enable:** The command is driven with FRAME# assertion, byte enables corresponding to supplied or requested data is driven on following clocks. |
| FRAME# | I/O 5V | **Frame:** Assertion indicates the address phase of a PCI transfer. Negation indicates that one more data transfer is desired by the cycle initiator. |
| DEVSEL# | I/O 5V | **Device Select:** This signal is driven by the TXC when a PCI initiator is attempting to access DRAM. DEVSEL# is asserted at medium decode time. |
| IRDY# | I/O 5V | **Initiator Ready:** Asserted when the initiator is ready for a data transfer. |
| TRDY# | I/O 5V | **Target Ready:** Asserted when the target is ready for a data transfer. |
| STOP# | I/O 5V | **Stop:** Asserted by the target to request the master to stop the current transaction. |
| LOCK# | I/O 5V | **Lock:** Used to establish, maintain, and release resource locks on PCI. |
| REQ[3:0]# | I 5V | **PCI Request:** PCI master requests for PCI. Weak external pull-up resistors are required on these signals. |
| GNT[3:0]# | O 5V | **PCI Grant:** Permission is given to the master to use PCI. Weak external pull-up resistors are required on these signals. |
| PHLD# | I 5V | **PCI Hold:** This signal comes from the expansion bridge. It is the bridge request for PCI. The PHLD# protocol supports passive release. A weak external pull-up resistor is required on this signal. |
| PHLDA# | O 5V | **PCI Hold Acknowledge:** This signal is driven by the TXC to grant PCI to the expansion bridge. The PHLDA# protocol supports passive release. A weak external pull-up resistor is required on this signal. |
| PAR | I/O 5V | **Parity:** A single parity bit is provided over AD[31:0] and C/BE[3:0]. |
| SERR# | O 5V | **SYSTEM ERROR:** The TXC asserts SERR# to signal a system error. A system error can be optionally generated for either single bit error (correctable) events or any ECC bit error (correctable or uncorrectable), or for parity error. |

**NOTES:**

All signals in the PCI interface conform to the PCI Rev 2.1 specification

## 2.5.    Clock, Reset, and Test

| Name | Type | Description |
|------|------|-------------|
| RST# | I 5V | **Reset:** When asserted this signal will asynchronously reset the TXC. The PCI signals will also tri-state compliant to PCI Rev 2.0 and 2.1 specifications. |
| TEST# | I 5V | **Test In:** NAND-tree mode is activated by driving this pin high when REQ [3:0]# pins are 0010. This signal requires an external pull-up resistor. |
| HCLKIN | I 5V | **Host Clock In:** This pin receives a buffered host clock. This clock is used by all of the TXC logic that is in the Host clock domain. |
| PCLKIN | I 5V | **PCI Clock In:** This pin receives a buffered divide-by-2 host clock. This clock is used by all of the TXC logic that is in the PCI clock domain. |
| VDD5V | I 5V | **VDD 5V reference:** This pin provides a 5V reference for all 5V I/O. It requires a 100 ohms series resistor and a 1.0 micro-farad cap to ground. |

## 3.0. REGISTER DESCRIPTION

The TXC contains two sets of software accessible registers (I/O Control and Configuration registers), accessed via the Host CPU I/O address space. The I/O Control registers control access to PCI configuration space. Configuration Registers reside in PCI configuration space and specify PCI configuration, DRAM configuration, cache configuration, operating parameters, and optional system features.

The TXC internal registers (both I/O Control and Configuration registers) are only accessible by the Host CPU and cannot be accessed by PCI masters. The registers can be accessed as Byte, Word (16-bit), or Dword (32-bit) quantities, with the exception of CONFADD which can only be accessed as a Dword. All multi-byte numeric fields use "little-endian" ordering (i.e., lower addresses contain the least significant parts of the field). The following nomenclature is used for access attributes.

**RO**     **Read Only**. If a register is read only, writes to this register have no effect.

**R/W**     **Read/Write**. A register with this attribute can be read and written.

**R/WC**     **Read/Write Clear**. A register bit with this attribute can be read and written. However, a write of 1 clears (sets to 0) the corresponding bit and a write of 0 has no effect.

Some of the TXC registers described in this section contain reserved bits. Software must deal correctly with fields that are reserved. On reads, software must use appropriate masks to extract the defined bits and not rely on reserved bits being any particular value. On writes, software must ensure that the values of reserved bit positions are preserved. That is, the values of reserved bit positions must first be read, merged with the new values for other bit positions and then written back.

In addition to reserved bits within a register, the TXC contains address locations in the PCI configuration space that are marked "Reserved" (Table 2). The TXC responds to accesses to these address locations by completing the Host cycle. Software should not write to reserved TXC configuration locations in the device-specific region (above address offset 3Fh).

During a hard reset (RST# asserted), the TXC sets its internal configuration registers to predetermined **default** states. The default state represents the minimum functionality feature set required to successfully bring up the system. Hence, it does not represent the optimal system configuration. It is the responsibility of the system initialization software (usually BIOS) to properly determine the DRAM configurations, cache configuration, operating parameters and optional system features that are applicable, and to program the TXC registers accordingly.

## 3.1. I/O Control Registers

The TXC contains two registers that reside in the CPU I/O address space—the Configuration Address (CONFADD) register and the Configuration Data (CONFDATA) register. The Configuration Address register enables/disables the configuration space and determines what portion of configuration space is visible through the Configuration Data window.

### 3.1.1. CONFADD—CONFIGURATION ADDRESS REGISTER

I/O Address:          0CF8h (Accessed as a Dword)
Default Value:        00000000h
Access:               Read/Write

CONFADD is a 32-bit register accessed only when referenced as a Dword. A Byte or Word reference will "pass through" the Configuration Address register to the PCI Bus. The CONFADD register contains the Bus Number, Device Number, Function Number, and Register Number for which a subsequent configuration access is intended.

| Bit | Descriptions |
|---|---|
| 31 | **Configuration Enable (CONE).** 1=Enable. 0=Disable. |
| 30:24 | **Reserved.** |
| 23:16 | **Bus Number (BUSNUM).** When BUSNUM is programmed to 00h, the target of the configuration cycle is either the TXC or the PCI Bus that is directly connected to the TXC, depending on the Device Number field. If the Bus Number is programmed to 00h and the TXC is not the target, a type 0 configuration cycle is generated on PCI. If the Bus Number is non-zero, a type 1 configuration cycle is generated on PCI with the Bus Number mapped to AD[23:16] during the address phase. |
| 15:11 | **Device Number (DEVNUM).** This field selects one agent on the PCI Bus selected by the Bus Number. During a Type 1 Configuration cycle, this field is mapped to AD[15:11]. During a Type 0 configuration cycle, this field is decoded and one of AD[31:11] is driven to 1. The TXC is always Device Number 0. |
| 10:8 | **Function Number (FUNCNUM).** This field is mapped to AD[10:8] during PCI configuration cycles. This allows the configuration registers of a particular function in a multi-function device to be accessed. The TXC responds to configuration cycles with a function number of 000b; all other function number values attempting access to the TXC (Device Number = 0, Bus Number = 0) generate a type 0 configuration cycle on the PCI Bus with no IDSEL asserted, which results in a master abort. |
| 7:2 | **Register Number (REGNUM).** This field selects one register within a particular bus, device, and function as specified by the other fields in the Configuration Address register. This field is mapped to AD[7:2] during PCI configuration cycles. |
| 1:0 | **Reserved.** |

### 3.1.2. CONFDATA—CONFIGURATION DATA REGISTER

I/O Address:          0CFCh
Default Value:        00000000h
Access:               Read/Write

CONFDATA is a 32-bit read/write window into configuration space. The portion of configuration space that is referenced by CONFDATA is determined by the contents of CONFADD.

| Bit | Descriptions |
|---|---|
| 31:0 | **Configuration Data Window (CDW).** If bit 31 of CONFADD is 1, any I/O reference in the CONFDATA I/O space is mapped to configuration space using the contents of CONFADD. |

## 3.2. PCI Configuration Space Mapped Registers

The PCI Bus defines a slot based "configuration space" that allows each device to contain up to 256 8-bit configuration registers. The PCI specification defines two bus cycles to access the PCI configuration space—**Configuration Read** and **Configuration Write**. While memory and I/O spaces are supported by the Pentium processor, configuration space is not supported. The PCI specification defines two mechanisms to access configuration space, Mechanism #1 and Mechanism #2. The TXC only supports Mechanism #1 (both type 0 and 1 accesses). Table 1 shows the TXC configuration space.

The configuration access mechanism makes use of the CONFADD register and CONFDATA register. To reference a configuration register, a Dword I/O write cycle is used to place a value into CONFADD that specifies the PCI Bus, the device on that bus, the function within the device, and a specific configuration register of the device function being accessed. CONFADD[31] must be 1 to enable a configuration cycle. Then, CONFDATA becomes a window onto four bytes of configuration space specified by the contents of CONFADD. Read/write accesses to CONFDATA generates a PCI configuration cycle to the address specified by CONFADD.

### 3.2.1. PCI CONFIGURATION ACCESS

**Type 0 Access:** If the Bus Number field of CONFADD is 0, a type 0 configuration cycle is generated on PCI. CONFADD[10:2] is mapped directly to AD[10:2]. The Device Number field of CONFADD is decoded onto AD[31:11]. The TXC is Device #0 and does not pass its configuration cycles to PCI. Thus, AD11 is never asserted. (For accesses to device #1, AD12 is asserted, etc., to Device #20 which asserts AD31.) Only one AD line is asserted at a time. All device numbers higher than 20 cause a type 0 configuration access with no IDSEL asserted, which results in a master abort.

**Type 1 Access:** If the Bus Number field of CONFADD is non-zero, a type 1 configuration cycle is generated on PCI. CONFADD[23:2] are mapped directly to AD[23:2]. AD[1:0] are driven to 01 to indicate a Type 1 Configuration cycle. All other lines are driven to 0.

**Table 1.  TXC Configuration Space**

| Address Offset | Register Symbol | Register Name | Access |
|---|---|---|---|
| 00–01h | VID | Vendor Identification | RO |
| 02–03h | DID | Device Identification | RO |
| 04–05h | PCICMD | PCI Command | R/W |
| 06–07h | PCISTS | PCI Status | RO, R/WC |
| 08h | RID | Revision Identification | RO |
| 09–0Bh | CLASSC | Class Code | RO |
| 0Ch | — | Reserved | — |
| 0Dh | MLT | Master Latency Timer | R/W |
| 0Eh | HEADT | Header Type | — |
| 0Fh | BIST | BIST | R/W |
| 10–4Fh | — | Reserved | — |
| 50h | PCON | PCI Control | R/W |
| 51h | — | Reserved | — |
| 52h | CC | Cache Control | R/W |

**Table 1.  TXC Configuration Space**

| Address Offset | Register Symbol | Register Name | Access |
|---|---|---|---|
| 53–55h | — | Reserved | — |
| 56h | DRAMEC | DRAM Extended Control | R/W |
| 57h | DRAMC | DRAM Control | R/W |
| 58h | DRAMT | DRAM Timing | R/W |
| 59–5Fh | PAM[6:0] | Programmable Attribute Map (7 registers) | R/W |
| 60–67h | DRB[7:0] | DRAM Row Boundary (8 registers) | R/W |
| 68h | DRT | DRAM Row Type | R/W |
| 69h | DRAT | DRAM Row Type | R/W |
| 70–71h | — | Reserved | — |
| 72h | SMRAM | System Management RAM Control | R/W |
| 73–8Fh | — | Reserved | — |
| 90h | ERRCMD | Error Command | R/W |
| 91h | ERRSTS | Error Status | RO, RWC |
| 92h | ERRSYN | Error Syndrome | RO |
| 93–FFh | — | Reserved | — |

### 3.2.2.    VID—VENDOR IDENTIFICATION REGISTER

Address Offset:          00–01h
Default Value:           8086h
Attribute:               Read Only

The VID register contains the vendor identification number. This 16-bit register combined with the Device Identification register uniquely identify any PCI device. Writes to this register have no effect.

| Bit | Description |
|---|---|
| 15:0 | **Vendor Identification Number:**  This is a 16-bit value assigned to Intel. |

15

### 3.2.3. DID—DEVICE IDENTIFICATION REGISTER

Address Offset:          02–03h
Default Value:           1250h
Attribute:               Read Only

This 16-bit register combined with the Vendor Identification register uniquely identifies any PCI device. Writes to this register have no effect.

| Bit | Description |
|-----|-------------|
| 15:0 | **Device Identification Number:** This is a 16-bit value assigned to the TXC. |

### 3.2.4. PCICMD—PCI COMMAND REGISTER

Address Offset:          04–05h
Default:                 06h
Access:                  Read/Write

This 16-bit register provides basic control over the TXC's ability to respond to PCI cycles. The PCICMD register in the TXC enables and disables the assertion of SERR# and PCI master accesses to main memory.

| Bit | Descriptions |
|-----|-------------|
| 15:10 | **Reserved**. |
| 9 | **Fast Back-to-Back (FB2B): (Not Implemented)** This bit is hardwired to 0. |
| 8 | **SERR# Enable (SERRE):** 1=Enable. 0=Disable. In systems that wish to report memory errors this bit should be set to 1 after memory has been scrubbed by BIOS |
| 7 | **Address/Data Stepping: (Not Implemented)** This bit is hardwired to 0. |
| 6 | **Parity Error Enable (PERRE): (Not Implemented)** This bit is hardwired to 0. |
| 5 | **Video Pallet Snooping (VPS): (Not Implemented)** This bit is hardwired to 0. |
| 4 | **Memory Write and Invalidate Enable (MWIE): (Not Implemented)** This bit is hardwired to 0. The TXC never uses the Memory Write and Invalidate PCI command. |
| 3 | **Special Cycle Enable (SCE): (Not Implemented)** This bit is hardwired to 0. The TXC does not respond to PCI special cycles. |
| 2 | **Bus Master Enable (BME): (Not Implemented)** This bit is hardwired to 1. The TXC does not support disabling of its bus master capability on the PCI Bus. |
| 1 | **Memory Access Enable (MAE):** 1=Enable PCI master access to main memory, if the PCI address selects enabled DRAM space; 0=Disable (TXC does not respond to main memory accesses). |
| 0 | **I/O Access Enable (IOAE): (Not Implemented)** This bit is hardwired to 0. The TXC does not respond to PCI I/O cycles. |

### 3.2.5. PCISTS—PCI STATUS REGISTER

Address Offset:          06–07h
Default Value:           0200h
Access:                  Read Only, Read/Write Clear

PCISTS is a 16-bit status register that reports the occurrence of a PCI master abort and PCI target abort. PCISTS also indicates the DEVSEL# timing that has been set by the TXC hardware. Bits [15:12,8] are read/write clear and bits [10:9] are read only.

| Bit | Descriptions |
|-----|--------------|
| 15 | **Detected Parity Error (DPE): (Not Implemented)**  This bit is hardwired to 0. |
| 14 | **Signaled System Error (SSE)—R/WC:** This bit is set to a 1 whenever the TXC signals an SERR#. |
| 13 | **Received Master Abort Status (RMAS)—R/WC:**  When the TXC terminates a Host-to-PCI transaction (TXC is a PCI master) with an unexpected master abort, this bit is set to 1. Note that master abort is the normal and expected termination of PCI special cycles. Software resets this bit to 0 by writing a 1 to it. |
| 12 | **Received Target Abort Status (RTAS)—R/WC:**  When a TXC-initiated PCI transaction is terminated with a target abort, RTAS is set to 1.  Software resets RTAS to 0 by writing a 1 to it. |
| 11 | **Signaled Target Abort Status (STAS):**  This bit is hardwired to 0, as the TXC never terminates a PCI cycle with a target abort. |
| 10:9 | **DEVSEL# Timing (DEVT)—RO:**  This 2-bit field indicates the timing of the DEVSEL# signal when the TXC responds as a target, and is hard-wired to the value 01b (medium) to indicate the slowest time that DEVSEL# is generated. |
| 8 | **Data Parity Detected (DPD):** This bit is hardwired to 0. |
| 7 | **Fast Back-to-Back (FB2B): (Not Implemented)** This bit is hardwired to 0. |
| 6 | **User Defined Format (UDF): (Not Implemented)**  This bit is hardwired to 0. The TXC does not contain any configurations that depend on the environment, such as network frequencies. |
| 5 | **66 MHz PCI Capable (66C): (Not Implemented)**  This bit is hardwired to 0. The TXC does not interface to 66 MHz PCI. |
| 4:0 | **Reserved**. |

### 3.2.6. RID—REVISION IDENTIFICATION REGISTER

Address Offset:          08h
Default Value:           Refer to product stepping information
Access:                  Read Only

This register contains the revision number of the TXC. These bits are read only and writes to this register have no effect.

| Bit | Description |
|-----|-------------|
| 7:0 | **Revision Identification Number:**  This is an 8-bit value that indicates the revision identification number for the TXC. |

| Bit | Description |
|-----|-------------|
| | A0 Stepping=00 |
| | A1 Stepping=01 |
| | A2 Stepping=02 |
| | A3 Stepping=03 |

### 3.2.7. CLASSC—CLASS CODE REGISTER

Address Offset:      09–0Bh
Default Value:      060000h
Access:      Read Only

This register contains the device programming interface information related to the Sub-Class Code and Base Class Code definition for the TXC. This register also contains the Base Class Code and the function sub-class in relation to the Base Class Code.

| Bit | Description |
|-----|-------------|
| 7:0 | **Programming Interface (PI):** 00h=No register-level programming interface defined. |
| 18:8 | **Sub-Class Code (SCC):** 00h=Host Bridge. |
| 23:16 | **Base Class Code (BASEC):** 06=Bridge device. |

### 3.2.8. MLT—MASTER LATENCY TIMER REGISTER

Address Offset:      0Dh
Default Value:      00h
Access:      Read/Write

MLT is an 8-bit register that controls the amount of time the TXC, as a bus master, can burst data on the PCI Bus. The Count Value is an 8-bit quantity. However, MLT[2:0] are hardwired to 0. MLT is used to guarantee the host CPU a minimum amount of the system resources.

| Bit | Description |
|-----|-------------|
| 7:3 | **Master Latency Timer Count Value.** The number of clocks programmed in the MLT represents the guaranteed time slice (measured in PCI clocks) allotted to the TXC, after which it must surrender the bus as soon as other PCI masters request the bus. The default value of MLT is 00h or 0 PCI clocks. |
| 2:0 | **Reserved.** |

### 3.2.9. HEADT—HEADER TYPE REGISTER

Address Offset:     0Eh
Default Value:      00h
Access:             Read Only

This register contains the Header Type of the TXC. This code is 00h indicating that the TXC's configuration space map follows the basic format. This register is read only.

| Bit | Description |
|-----|-------------|
| 7:0 | **Header Type (HTYPE):** 00h=Basic configuration space format. |

### 3.2.10. BIST—BIST REGISTER

Address Offset:     0Fh
Default Value:      00h
Access:             Read/Write

The Built In Self Test (BIST) function is not supported by the TXC. Writes to this register have no affect.

| Bit | Descriptions |
|-----|-------------|
| 7 | **BIST Supported:** This read only bit is always set to 0, disabling the BIST function. Writes to this bit position have no effect. |
| 6 | **Start BIST:** This function is not supported and writes have no effect. |
| 5:4 | **Reserved**. |
| 3:0 | **Completion Code:** This read only field always returns 0 when read and writes have no effect. |

### 3.2.11. ACON—ARBITRATION CONTROL

Address Offset:     4Fh
Default Value:      00h
Access:             Read/Write
Size:               8 bits

The ACON register enables and disables features related to the PCI arbitration and PCI 2.1 compliance.

| Bit | Descriptions |
|-----|-------------|
| 7 | **Extended CPU-toPIIX PHLDA# Signalling Enable (XPLDE):** When XPLDE = '1', the TXC adds the following additional signalling to signal PHLDA# (ie. In addition to the normal CPU/PIIX PHOLD/PHLDA# protocol); <br><br> 1) Whenever the TXC begins a PCI read/write transation, it will assert PHLDA# for 1 PCLK within the address phase of the transaction. <br><br> 2) If the CPU is attempting a LOCKed cycle AND LOCK has been established (ie. PLOCK# was seen deasserted in address phase), the PHLDA# remains asserted for one additional clock following the address phase. |
| 6:3 | **Reserved** |

19

| Bit | Descriptions |
|-----|--------------|
| 2 | **Enable Retry on First Transfer Disable (ERFD):** When ERFD=0 and GT2E=1, the TXC will retry a PCI cycle when the latency to the first data transfer is about to exceed 32 PCI clocks. When ERFD=1 or GT2E=0, the TXC will complete the first data transfer regardless of latency. This bit should be set to 0 for normal operation unless ECC is enabled, in which case this bit should be set to 1. |
| 1:0 | **Reserved.** |

### 3.2.12. PCON—PCI CONTROL

Address Offset:     50h
Default Value:      00h
Access:             Read/Write

The PCON register enables and disables features related to the PCI unit operation not already covered in the PCI required configuration space.

| Bit | Descriptions |
|-----|--------------|
| 7 | **DRAM ECC/Parity Select (DEPS).**  1=ECC. 0=Parity (default). Note that this bit also adjusts the DRAM timings as documented in the DRAM section to account for the ECC operation. |
| 6 | **ECC TEST Enable (ETE).**  1=ECC Test Mode. 0=Normal mode (default). When set, The TXC handles subsequent cycles to DRAM. |
| 5 | **Shutdown to Port 92 (SP92).**  When SP92=0, the TXC forwards a Shutdown special cycle from the host bus to PCI. When SP92=1, the TXC writes 01h to I/O address 92 on PCI in response to a Shutdown special cycle on the host bus. |
| 4 | **Dual Processor NA# Enable (DPNAE).**  When DPNAE=0, the 82430FX PCIset NA# policies are used. When DPNAE=1, the NA# assertion policies are modified to prevent deadlocks in dual-processor PCI 2.1 systems. DPNAE should be correctly set by BIOS prior to the L1 cache being enabled. |
| 3 | **Peer Concurrency Enable (PCE).** When PCE=1, the CPU will be allowed to run DRAM/L2 cycles when non-PHLD PCI masters are running non-locked cycles targeting PCI peer devices. CPU-to-PCI cycles will be blocked (BRDY# stalled) during these peer cycles. When PCE=0, the CPU will be held off the bus during PCI peer cycles. This bit should be set to 1 for normal operation. |
| 2 | **SERR# Output Type (SOT).** 1=SERR# is a normal output actively driven high when negated. 0(default)=SERR# is an open drain output, compatible to the PCI SERR# signal.  Note that SOT should not be programmed to a 1 if SERR# is connected to the PCI bus. SOT should only be programmed to a 1 in systems where SERR# is used as a sideband error signal (e.g. in ECC systems). |
| 1 | **Reserved.** |
| 0 | **Global TXC Enable (GTE).**  This bit must be set to 1 to enable all new 82439HX features. |

#### 3.2.13. CC—CACHE CONTROL REGISTER

Address Offset:       52h
Default Value:        SSSS0010 (S = Strapping option)
Access:               Read/Write

This 8-bit register defines the secondary cache operations. The CC register enables and disables the second level cache, adjusts cache size, selects the cache write policy, selects the caching policy when CACHE# is negated on reads, informs the TXC how the SRAMs are connected, and defines the cache SRAM type. After a hard reset, CC[7:4] reflect the signal levels on the Host address lines A[31:28].

| Bit | Description |
|---|---|
| 7:6 | **Secondary Cache Size (SCS):** This field reflects the inverted signal level on the A[31:30] pins at the rising edge of the RESET signal. The default value can be overwritten with subsequent writes to the CC register. The options are:<br><br>**Bits[7:6]    Secondary Cache Size**<br><br>0   0      Cache not populated<br>0   1      256 Kbytes<br>1   0      512 Kbytes<br>1   1      Reserved<br><div align="center">**NOTE**</div><br>1.   When SCS=00, the secondary cache is disabled.<br><br>2.   To enable the L2 cache, SCS must be non-zero and the FLCE bit must be 1. |
| 5:4 | **SRAM Type (SRAMT):** This field reflects the inverted signal level on the A[29:28] pins at the rising edge of the RESET signal. The RESET values can be overwritten with subsequent writes to the CC register. The options are:<br><br>**Bits[5:4]    SRAM Type**<br><br>0   0      One bank of Pipelined Burst<br>0   1      Reserved<br>1   0      Reserved<br>1   1      Two banks of Pipelined Burst |
| 3 | **NA Disable (NAD):** When NAD=1, the TXC's NA# pin is never asserted. When NAD=0, NA# assertion is dependent on the cache type and size selected (via SRAMT, SCS). Note that NAD must be set to 1 if the NA pin of the TXC is not connected to the processor. This bit should be set to 0 for normal operation in systems that connect NA# to the processor. |
| 2 | **Extended Cacheability Enable (ECE).** 1=512-Mbyte limit (TIO[10:8] lines are enabled). 0=64-Mbyte limit (TIO[10:8] are disabled). |
| 1 | **Secondary Cache Force Miss or Invalidate (SCFMI):** When SCFMI=1, the L2 hit/miss detection is disabled, and all tag lookups result in a miss. If the L2 is enabled, the cycle is processed as a miss. If the L2 is populated but disabled (FLCE=0), then when SCFMI is set to a 1, any CPU read cycle will invalidate the selected tag entry. When SCFMI=0, normal L2 cache hit/miss detection and cycle processing occurs.<br><br>Software can flush the cache (cause all modified lines to be written back to DRAM) by setting SCFMI to a 1 with L2 enabled (non-zero SCS, FLCE=1) immediately after reading twice the size of the L2.<div align="center">**NOTE**</div>For a 512-Kbyte cache, software must read 1 Mbyte of non-overlapping, cacheable 512-Kbyte regions from the memory before setting the SCFMI bit. |

| Bit | Description |
|-----|-------------|
| 0 | **First Level Cache Enable (FLCE):** FLCE enables/disables the first level cache. When FLCE=1, the TXC responds to CPU cycles with KEN# asserted for cacheable memory cycles. When FLCE=0, KEN# is always negated and new cache line fills to either the first level or second level cache are prevented. Note that, when FLCE=1 and SCFMI=1, writes to the cache are also forced as misses. Thus, it is possible to create incoherent data between main memory and the L2 cache. A summary of FLCE/SCFMI bit interactions is as follows:<br><br>**FLCE   SCFMI   L2 Cache Result**<br>0          0          Disabled<br>0          1          Disabled; tag invalidate on reads<br>1          0          Normal L2 cache operation (dependent on SCS)<br>1          1          Enabled; miss forced on reads/writes |

### 3.2.14.   DRAMEC—DRAM EXTENDED CONTROL REGISTER

Address Offset:         56h
Default Value:          00h
Access:                     Read/Write

This 8-bit register contains additional controls for main memory DRAM operating modes and features.

| Bit | Descriptions |
|-----|--------------|
| 7:5 | **Reserved.** |
| 4 | **Speculative Leadoff Enable (SLE).** When SLE=1, the DRAM controller read request is presented before the final memory target (cache/DRAM/PCI) has been decoded by the TXC. This results in a 1 HCLK pull-in for all read leadoff latencies. Note that if the cycle does not actually target DRAM, the DRAM cycle is later terminated. |
| 3 | **Turn-around Insertion Enable (TIEN):** When TIEN=1, the TXC inserts one extra clock of turn-around on the MD lines after asserting MWE# before enabling the TXC MD buffers. When TIEN = 0, the TXC controls back-to-back DRAM cycles just like the 82430FX PCIset. |
| 2:1 | **Memory Address Drive Strength (MAD):** This field controls the strength of the output buffers driving the MA and MWE# pins.<br><br>**Bit 2      Bit 1      MAA[1:0]/MAB[1:0]      MA[11:2]/MWE#**<br>0             0             8 mA                               8 mA<br>0             1             8 mA                               12 mA<br>1             0             12 mA                             8 mA<br>1             1             12 mA                             12 mA |
| 0 | **64 Mbit Mode Enable (64ME):** When 64ME=1, the 64 Mbit density SIMMs are support by changing the MA address mux selection on row bit 11. Refer to the DRAM Interface section for more details. |

### 3.2.15. DRAMC—DRAM CONTROL REGISTER

Address Offset:      57h
Default Value:      01h
Access:      Read/Write

This 8-bit register controls main memory DRAM operating modes and features.

| Bit | Description |
|-----|-------------|
| 7:6 | **Hole Enable (HEN):** This field enables a memory hole in DRAM space. CPU cycles matching an enabled hole are passed on to PCI. PCI cycles matching an enabled hole are ignored by the TXC (no DEVSEL#). Note that a selected hole is not remapped.<br><br>**Bits[7:6]**      **Hole Enabled**<br><br>00      None<br>01      512 Kbytes – 640 Kbytes<br>10      15 - 16 Mbytes<br>11      Reserved |
| 5:4 | **Reserved.** |
| 3 | **EDO Detect Mode Enable (EDME):** This bit, if set to a 1, enables a special timing mode for BIOS to detect EDO DRAM type on a bank-by-bank basis. Once all DRAM row banks have been tested for EDO, the EDME bit should be set to 0. Otherwise, performance will be seriously impacted. |
| 2:0 | **DRAM Refresh Rate (DRR):** The DRAM refresh rate is adjusted according to the frequency selected by this field. Note that refresh is also disabled via this field, and that disabling refresh results in the eventual loss of DRAM data.<br><br>**Bits[2:0]**      **Host Bus Frequency**<br><br>000      Refresh Disabled<br>001      50 MHz<br>010      60 MHz<br>011      66 MHz<br>1XX      Reserved<br><br>DRR0 is initialized to the inverted signal level on A27 at the rising edge of reset. A jumpered pull-up resistor option on A27 allows the BIOS to read DRR0 to determine 60 MHz bus operation. Otherwise, A27 should be externally pulled down for 66 MHz default. Subsequent writes to this field will override the reset strap value. |

### 3.2.16. DRAMT—DRAM TIMING REGISTER

Address Offset:      58h
Default Value:      00h
Access:      Read/Write

This 8-bit register controls main memory DRAM timings.

Systems that support ECC must set read and write burst rates to x333 or slower. Additionally, x222 timings are only supported for systems designed for up to 4 DRAM SIMMS and which do not use external MA buffers. Only 60ns, or faster EDO DRAMs will meet x222 EDO timings.

| Bit | Description |
|-----|-------------|
| 7 | **Turbo Read Leadoff (TRL):** This bit, when set, enables bypass of the first input register in the DRAM data pipeline, resulting in a 1 HCLK pull-in of all read  leadoff timings. TRL can only be set to 1 in a cacheless configuration.<br><div align="center">**NOTE**</div><br>It is illegal to enable this feature if ERRCMD[1:0] is not 00. |
| 6:5 | **DRAM Read Burst Timing (DRBT):**  The DRAM read burst timings are controlled by the DRBT field.  Slower rates may be required in certain system designs to support layouts with longer trace lengths or  slower DRAMs. Most system designs will be able to use one of the faster burst mode timings. The timing used depends on the type of DRAM on a per-bank basis, as indicated by the DRT register.<br><br>**DRBT**　**EDO Burst Rate**　**Standard Page Mode Rate**<br>00　x444　x444<br>01　x333　x444<br>10　x222　x333<br>11　Reserved　Reserved |
| 4:3 | **DRAM Write Burst Timing (DWBT):**  The DRAM write burst timings are controlled by the DWBT field. Slower rates may be required in certain system designs to support layouts with longer trace lengths or  slower DRAMs. Most system designs will be able to use one of the faster burst mode timings.<br><br>**DWBT**　**Burst Rate**<br>00　x444<br>01　x333<br>10　x222<br>11　Reserved |
| 2 | **Fast RAS to CAS Delay (FRCD):**  The DRAM row miss leadoff timings are controlled by the FRCD bit.  When FRCD=1, the RAS active to CAS active delay is 2 clocks. When FRCD=0, the timing is 3 host clocks. Slower rates may be required in certain system designs to support layouts with longer trace lengths or  slower DRAMs. If DLT=x0, FRCD must be set to 0. If DLT=x1, either setting of FRCD is legal. |
| 1:0 | **DRAM Leadoff Timing (DLT):**  The DRAM leadoff timings are controlled by the DLT bits. Slower rates may be required in certain system designs to support layouts with longer trace lengths or  slower DRAMs. The page hit leadoff timings are summarized below:  Page miss and bank miss timings are also affected (DLT controls the MA setup to the first CAS# assertion).<br><br>**DLT**　**Read Leadoff**　**Write Leadoff**　**RAS# Precharge**　**Refresh RAS Assertion**<br>00　7　6　3　4<br>01　6　5　3　4<br>10　7　6　4　5<br>11　6　5　4　5<br><div align="center">**NOTES**</div><br>1.　The DLT field and the FRCD bit have cumulative affects (i.e., setting DLT0=0 and FRCD=0 results in two additional clocks between RAS# assertion and CAS# assertion).<br><br>2.　The SLE and TRL fields have subtractive effects for reads only (i.e.,  setting SLE=1 and TRL=1 results in two less clocks  from the read leadoff count).<br><br>3.　If ECC is enabled, the combination of slow leadoffs (DLT=x0) and x222 write burst timings (DWBT=10) does not guarantee a 2-clock MWE# setup to CAS#. In all other cases, when DLT=x0, a 2-clock MWE# setup to CAS# is guaranteed. |

intel®

### 3.2.17. PAM—PROGRAMMABLE ATTRIBUTE MAP REGISTERS (PAM[6:0])

Address Offset:       PAM0 (59h) — PAM6 (5Fh)
Default Value:        00h
Attribute:            Read/Write

The TXC allows programmable memory and cacheability attributes on 14 memory segments of various sizes in the 640-Kbyte to 1-Mbyte address range. The PAM registers support these features. Three bits are used to specify L1 cacheability and memory attributes for each memory segment. These attributes are:

**Read Enable (RE)**       When RE=1, the CPU read accesses to the corresponding memory segment are directed to main memory. Conversely, when RE=0, the CPU read accesses are directed to PCI.

**Write Enable (WE)**      When WE=1, the CPU write accesses to the corresponding memory segment are directed to main memory. Conversely, when WE=0, the CPU write accesses are directed to PCI.

**Cache Enable (CE)**      When CE=1, the corresponding memory segment is L1 cacheable. CE must not be set to 1 when RE is reset to 0 for any particular memory segment. When CE=1 and WE=0, the corresponding memory segment is cached in the first level cache only on CPU code read cycles.

The RE and WE attributes permit a memory segment to be Read Only, Write Only, Read/Write, or disabled. For example, if a memory segment has RE=1 and WE=0, the segment is Read Only.

**Table 2.  Attribute Definition**

| Read/Write Attribute | Definition |
|---|---|
| Read Only | Read cycles: CPU cycles are serviced by the DRAM in a normal manner. |
| | Write cycles: CPU initiated write cycles are ignored by the DRAM interface as well as the cache. Instead, the cycles are passed to PCI for termination. |
| | Areas marked as Read Only are L1 cacheable for Code accesses only.  These regions are not cached in the second level cache. |
| Write Only | Read cycles: All read cycles are ignored by the DRAM interface as well as the second level cache. CPU-initiated read cycles are passed onto PCI for termination. The write only state can be used while copying the contents of a ROM, accessible on PCI, to main memory for shadowing, as in the case of BIOS shadowing. |
| | Write cycles: CPU write cycles are serviced by the DRAM and L2 cache in a normal manner. |
| Read/Write | This is the normal operating mode of main memory. Both read and write cycles from the CPU and PCI are serviced by the DRAM and L2 cache interface. |
| Disabled | All read and write cycles to this area are ignored by the DRAM and cache interface. These cycles are forwarded to PCI for termination. |

Each PAM register controls two regions, typically 16 Kbytes in size. Each of these regions has a 4-bit field. The four bits that control each region have the same encoding.

PCI master access to DRAM space is also controlled by the PAM registers. If the PAM programming indicates a region is writeable, PCI master writes are accepted (DEVSEL# generated). If the PAM programming indicates a region is readable, PCI master reads are accepted. If a PCI write to a non-writeable DRAM region, or a PCI read to a non-readable DRAM region is seen, the TXC does not accept the cycle (DEVSEL# not asserted). PCI master accesses to enable memory hole regions are not accepted.

As an example, consider a BIOS that is implemented on the expansion bus. During the initialization process the BIOS can be shadowed in main memory to increase the system performance. When a BIOS is shadowed in main memory, it should be copied to the same address location. To shadow the BIOS, the attributes for that address range should be set to write only. The BIOS is shadowed by first doing a read of that address. This read is forwarded to the expansion bus. The CPU then does a write of the same address, which is directed to main memory. After the BIOS is shadowed, the attributes for that memory area are set to read only so that all writes are forwarded to the expansion bus.

**Table 3.  PAM Registers and Associated Memory Segments**

| PAM Reg | Attribute Bits | | | | Memory Segment | Comments | Offset |
|---------|-----|-----|-----|-----|----------------|----------|--------|
|         | 7,3 | 6,2 | 5,1 | 4,0 |                |          |        |
| PAM0[3:0] | Reserved | | | |                |          | 59h |
| PAM0[7:4] | R | CE | WE | RE | 0F0000h–0FFFFFh | BIOS Area | 59h |
| PAM1[3:0] | R | CE | WE | RE | 0C0000h–0C3FFFh | ISA Add-on BIOS | 5Ah |
| PAM1[7:4] | R | CE | WE | RE | 0C4000h–0C7FFFh | ISA Add-on BIOS | 5Ah |
| PAM2[3:0] | R | CE | WE | RE | 0C8000h–0CBFFFh | ISA Add-on BIOS | 5Bh |
| PAM2[7:4] | R | CE | WE | RE | 0CC000h–0CFFFFh | ISA Add-on BIOS | 5Bh |
| PAM3[3:0] | R | CE | WE | RE | 0D0000h–0D3FFFh | ISA Add-on BIOS | 5Ch |
| PAM3[7:4] | R | CE | WE | RE | 0D4000h–0D7FFFh | ISA Add-on BIOS | 5Ch |
| PAM4[3:0] | R | CE | WE | RE | 0D8000h–0DBFFFh | ISA Add-on BIOS | 5Dh |
| PAM4[7:4] | R | CE | WE | RE | 0DC000h–0DFFFFh | ISA Add-on BIOS | 5Dh |
| PAM5[3:0] | R | CE | WE | RE | 0E0000h–0E3FFFh | BIOS Extension | 5Eh |
| PAM5[7:4] | R | CE | WE | RE | 0E4000h–0E7FFFh | BIOS Extension | 5Eh |
| PAM6[3:0] | R | CE | WE | RE | 0E8000h–0EBFFFh | BIOS Extension | 5Fh |
| PAM6[7:4] | R | CE | WE | RE | 0EC000h–0EFFFFh | BIOS Extension | 5Fh |

**NOTES:**

The CE bit should not be changed while the L2 cache is enabled.

**DOS Application Area (00000–9FFFh)**
Read, write, and cacheability attributes are always enabled and are not programmable for the 0–640-Kbyte DOS application region.

**Video Buffer Area (A0000–BFFFFh)**
This 128-Kbyte area is not controlled by attribute bits. CPU-initiated cycles in this region are always forwarded to PCI for termination. This area is not cacheable.

**Expansion Area (C0000–DFFFFh)**
This 128-Kbyte area is divided into eight 16-Kbyte segments. Each segment can be assigned one of four Read/Write states: read-only, write-only, read/write, or disabled Memory that is disabled is not remapped. Cacheability status can also be specified for each segment.

**Extended System BIOS Area (E0000–EFFFFh)**
This 64-Kbyte area is divided into four 16-Kbyte segments. Each segment can be assigned independent cacheability, read, and write attributes. Memory segments that are disabled are not remapped elsewhere.

**System BIOS Area (F0000−FFFFFh)**

This area is a single 64-Kbyte segment. This segment can be assigned cacheability, read, and write attributes. When disabled, this segment is not remapped.

**Extended Memory Area (100000−FFFFFFFFh)**

The extended memory area can be split into several parts;

- Flash BIOS area from 4 Gbytes to 4 Gbytes − 512 Kbytes (aliased on ISA at 16 Mbyte − 15.5 Mbyte)

- DRAM Memory from 1 Mbyte to a maximum of 512 Mbytes

- PCI Memory space from the top of DRAM to 4 Gbytes − 512 Kbyte

On power-up or reset the CPU vectors to the Flash BIOS area, mapped in the range of 4 Gbyte to 4 Gbyte minus 512 Kbyte. This area is physically mapped on the expansion bus. Since these addresses are in the upper 4-Gbyte range, the request is directed to PCI.

The DRAM memory space can occupy extended memory from a minimum of 1 Mbyte up to 512 Mbytes. This memory is cacheable.

PCI memory space from the top of main memory to 4 Gbytes is always non-cacheable.

### 3.2.18.    DRB—DRAM ROW BOUNDARY REGISTERS

Address Offset:        DRB0 (60h) — DRB7 (67h)
Default Value:        02h
Access:        Read/Write

The TXC supports 8 rows of DRAM. Each row is 64 bits wide. The DRAM Row Boundary registers define upper and lower addresses for each DRAM row. Contents of these 8-bit registers represent the boundary addresses in 4-Mbyte granularity.

    DRB0 = Total amount of memory in row 0  (in 4 Mbytes)
    DRB1 = Total amount of memory in row 0 + row 1 (in 4 Mbytes)
    DRB2 = Total amount of memory in row 0 + row 1 + row 2 (in 4 Mbytes)
    DRB3 = Total amount of memory in row 0 + row 1 + row 2 + row 3 (in 4 Mbytes)
    DRB4 = Total amount of memory in row 0 + row 1 + row 2 + row 3 + row 4 (in 4 Mbytes)
    etc ...

The DRAM array can be configured with 512 K , 1 M, 4 M, or 16 M deep by 32 or 36-bit wide SIMMs. Each register defines an address range that causes a particular RAS# line to be asserted (e.g., if the first DRAM row is 8 Mbytes in size, accesses within the 0 to 8-Mbyte range cause RAS0# to be asserted). The DRB registers are programmed with an 8-bit upper address limit value. This upper address limit is compared to A[29:22] of the Host address bus, for each row, to determine if DRAM is being targeted.

| Bit | Description |
|-----|-------------|
| 7:0 | **Row Boundary Address:**  This 8-bit value is compared against address lines A[29:22] to determine the upper address limit of a particular row (i.e., DRB minus previous DRB = row size). |

**Row Boundary Address**

These 8-bit values represent the upper address limits of the 8 rows (i.e., this row minus previous row = row size). Unpopulated rows have a value equal to the previous row (row size = 0). DRB7 reflects the maximum amount of DRAM in the system.  The top of memory is determined by the value written into DRB7. If DRB7 is greater than 512 Mbytes, then 512 Mbytes of DRAM are available.

As an example of a general purpose configuration where 4 physical rows are configured for either single-sided or double-sided SIMMs, the memory array would be configured as shown in Figure 2. In this configuration, the TXC drives two RAS# signals directly to the SIMM rows. If single-sided SIMMs are

populated, the even RAS# signal is used and the odd RAS# is not connected. If double-sided SIMMs are used, both RAS# signals are used.
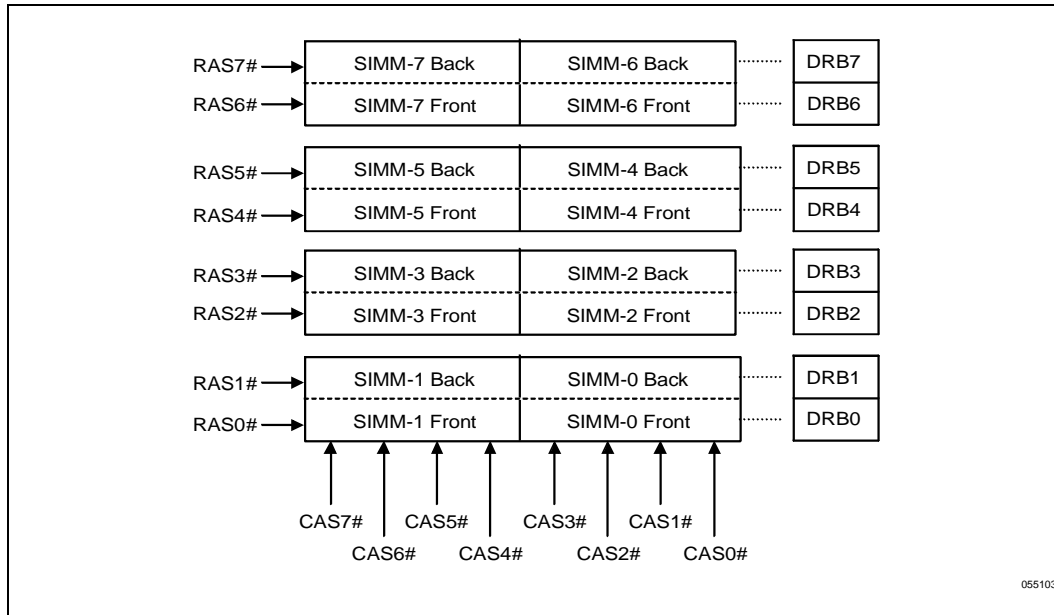


**Figure 2.  SIMMs and Corresponding DRB Registers**

The following 2 examples describe how the DRB registers are programmed for cases of single-sided and double-sided SIMMs on a motherboard having a total of four 8-byte or eight 4-byte SIMM sockets.

**Example #1**
The memory array is populated with four single-sided 1MB x 32 SIMMs, a total of 16 Mbytes of DRAM. Two SIMMs are required for each populated row making each populated row 8 Mbytes in size.

    DRB0 = 02h        populated (2 SIMMs, 8 Mbytes this row)
    DRB1 = 02h        empty row
    DRB2 = 04h        populated (2 SIMMs, 8 Mbytes this row)
    DRB3 = 04h        empty row
    DRB4 = 04h        empty row
    DRB5 = 04h        empty row
    DRB6 = 04h        empty row
    DRB7 = 04h        empty row

**Example #2**

In this example, the memory array is populated with two 2Mx32 double-sided SIMMs (one row), and four 4Mx32 single-sided SIMMs (two rows), for a total of 80 Mbytes of DRAM. The DRB registers are programmed as follows:

| | |
|---|---|
| DRB0 = 02h | populated with 8 Mbytes (half of double-sided SIMMs) |
| DRB1 = 04h | populated with 8 Mbytes (other half of the double-sided SIMMs) |
| DRB2 = 0Ch | populated with 32 Mbytes (two single-sided SIMMs) |
| DRB3 = 0Ch | empty row |
| DRB4 = 14h | populated with 32 Mbytes (other 32 Mbytes of single-sided SIMMs) |
| DRB5 = 14h | empty row |
| DRB6 = 14h | empty row |
| DRB7 = 14h | empty row |

### 3.2.19. DRT—DRAM ROW TYPE REGISTER

| | |
|---|---|
| Address Offset: | 68h |
| Default Value: | 00h |
| Access: | Read/Write |

This 8-bit register identifies the type of DRAM (EDO or page mode) used in each row, and should be programmed by BIOS for optimum performance if EDO DRAMs are used. The hardware uses these bits to determine the correct cycle timing to use before a DRAM cycle is run.

| Bit | Description |
|---|---|
| 7:0 | **DRAM Row Type (DRT):** Each bit in this register corresponds to the DRAM row identified by the corresponding DRB register. When the DRT bit for a DRB register is programmed to a 0, then page mode DRAM timings are used for that bank. If the DRT bit for a DRB register is programmed to a 1, then EDO DRAM timings are used for that bank. DRT[0] corresponds to row 0, DRT[1] to row 1, etc. |

### 3.2.20. SMRAM—SYSTEM MANAGEMENT RAM CONTROL REGISTER

| | |
|---|---|
| Address Offset: | 72h |
| Default Value: | 02h |
| Access: | Read/Write |

The System Management RAM Control register controls how accesses to this space are treated. The Open, Close, and Lock SMRAM Space bits function only when the SMRAM enable bit is set to a 1. Also, the OPEN bit should be reset before the LOCK bit is set.

| Bit | Description |
|---|---|
| 7 | **Reserved.** |
| 6 | **SMM Space Open (DOPEN):** When DOPEN=1 and DLCK=0 SMM space DRAM is made visible even when SMIACT# is negated. This is intended to help BIOS initialize SMM space. Software should ensure that DOPEN=1 is mutually exclusive with DCLS=1. When DLCK is set to a 1, DOPEN is reset to 0 and becomes read only. |

| Bit | Description |
|-----|-------------|
| 5 | **SMM Space Closed (DCLS):** When DCLS=1, SMM space DRAM is not accessible to data references, even if SMIACT# is asserted. Code references may still access SMM space DRAM. This allows SMM software to reference "through" SMM space to update the display even when SMM space is mapped over the VGA range. Software should ensure that DOPEN=1 is mutually exclusive with DCLS=1. |
| 4 | **SMM Space Locked (DLCK):** When DLCK is set to 1, DOPEN is reset to 0 and both DLCK and DOPEN become read only. DLCK can be set to 1 via a normal configuration space write but can only be cleared by a power-on reset. The combination of DLCK and DOPEN provide convenience with security. The BIOS can use the DOPEN function to initialize SMM space and then use DLCK to "lock down" SMM space in the future so that no application software (or BIOS itself) can violate the integrity of SMM space, even if the program has knowledge of the DOPEN function. |
| 3 | **SMRAM Enable (SMRAME):** If set to a 1, the SMRAM function is enabled, providing 128 Kbytes of DRAM accessible at the A0000h address while in SMM (ADS# with SMIACT#). Refer to the section on SMM for more details. |
| 2:0 | **SMM Space Base Segment (DBASESEG):** This field programs the location of SMM space. "SMM DRAM" is not remapped. It is simply "made visible" if the conditions are right to access SMM space, otherwise the access is forwarded to PCI. DBASESEG = 010 is the only allowable setting, which selects the SMM space as A0000−BFFFFh. All other values are reserved. PCI initiators are not allowed access to SMM space. |

Table 5 summarizes the operation of SMRAM space cycles targeting SMI space addresses (A-segment):

**Table 4. SMRAM Space Cycles**

| SMRAME | DLCK | DCLS | DOPEN | SMIACT# | Code Fetch | Data Reference |
|--------|------|------|-------|---------|-----------|----------------|
| 0 | X | X | X | X | PCI | PCI |
| 1 | 0 | 0 | 0 | 0 | DRAM | DRAM |
| 1 | 0 | X | 0 | 1 | PCI | PCI |
| 1 | 0 | 0 | 1 | X | DRAM | DRAM |
| 1 | 0 | 1 | 0 | 0 | DRAM | PCI |
| 1 | 0 | 1 | 1 | X | INVALID | INVALID |
| 1 | 1 | 0 | X | 0 | DRAM | DRAM |
| 1 | 1 | X | X | 1 | PCI | PCI |
| 1 | 1 | 1 | X | 0 | DRAM | PCI |

### 3.2.21. ERRCMD—ERROR COMMAND REGISTER

Address Offset:     90h
Default Value:      00h
Access:             Read/Write

This 8-bit register controls the operation of the Error Detection and Correction feature.

| Bit | Description |
|-----|-------------|
| 7 | **SERR# Duration (SED).** This bit determines the duration that SERR# is asserted when the TXC signals an error. The TXC signals an error if PCICMD[SERRE]=1 and ERRCMD[SMUE]=1 when an uncorrectable ECC or parity error is detected, or PCICMD[SERRE]=1 and ERRCMD[SSCE]=1 when a correctable ECC error is detected, or PCICMD[SERRE]=1 and ERRCMD[SMUE]=1 when a correctable ECC error is detected. <br><br> When SED=0 and the conditions above are met, SERR# is asserted for 1 PCI clock.  This setting, also referred to as pulse mode, is used to report memory errors via NMI. <br><br> When SED=1, once SERR# becomes asserted, it will remain asserted until the error flags (MEF, SET) with a respective enable set (SMUE, SSCE) are cleared (i.e., MEF*SMUE + SEF*SSCE=0). This setting, also referred to as level mode, is used to report memory errors via an SMI event or APIC interrupt. <br><br> Note that the SED setting does not affect software-generated error signaling that is always a pulse mode (see MEF description). <br><br> This bit should be programmed to the same value as PCON[SOT, bit 2}. |
| 6:3 | **Reserved.** |
| 2 | **Bad PAR on Multiple-Bit Uncorrectable Error (BPARE).**  When BPARE=1, the TXC forces incorrect PCI parity on PCI read data starting from the time that an uncorrectable DRAM read error occurs (either a multi-bit ECC or parity error) during a PCI read cycle until the end of that current cycle.  Subsequent PCI read cycles will have correct parity presented.  Note that it is the uncorrectable read *event* that begins the bad parity forcing, *not* the state of the MEF flag.  Also, bad parity is immediately delivered to PCI, even if the error event occurs during a read prefetch for data that the PCI master may ultimately not request in that cycle. *For systems not supporting either ECC or parity, this bit must be set to 0.* |
| 1 | **SERR# on Multiple-Bit Uncorrectable Error (SMUE).** When SMUE=1, the TXC asserts SERR# when it detects a multi-bit ECC or parity  error. When SMUE=0, the TXC does not assert SERR# on multi-bit or parity errors. The SERR# Enable bit (SERRE) in the PCICMD register must also be set to signal the SERR#. *For systems not supporting either ECC or parity, this bit must be set to 0.* |
| 0 | **SERR# on Single-Bit Correctable Error (SSCE).**  When SSCE=1, the TXC asserts SERR# when it detects a single-bit ECC error. When SSCE=0, the TXC will not assert SERR# on single-bit errors.  The SERR# Enable bit (SERRE) in the PCICMD register must also be set to signal the SERR#. |

Table 5 shows the SED and SOT configurations:

**Table 5.  SED/SOT Configurations**

| SED | SOT | System Configuration |
|-----|-----|----------------------|
| 0 | 0 | SERR# output connected to PCI, DRAM errors signaled via NMI. |
| 0 | 1 | Reserved |
| 1 | 0 | Reserved |
| 1 | 1 | SERR# output connected to external logic, DRAM errors signaled via SMI. |

### 3.2.22.    ERRSTS—ERROR STATUS REGISTER

Address Offset:          91h
Default Value:           00h
Access:                  Read Only / Read Write Clear

This 8-bit register contains status of the operation of the Error Detection and Correction feature.

| Bit | Description |
|-----|-------------|
| 7:5 | **Multi-Bit First Row Error (MBFRE)**  This field is the encoded value of the DRAM row associated with the multi-bit error.  When an error is detected, this field is updated and the MEF bit is set. This field will be locked (no further updates) until the MEF flag has been reset. When MEF=0, the value in this field is undefined. |
| 4 | **Multi-Bit (uncorrectable) Error Flag (MEF).** The TXC sets this bit to 1 when  an uncorrectable multi-bit errors  is detected with ECC enabled. If SERR# signaling is enabled by the ERRCMD[SMUE] bit, and PCICMD[SERRE] is set, then the error is reported via the SERR# pin. SERR# must be disabled, by setting PCICMD[SERRE]='0' before any write to the ERRSTS register except the ones which are done for the explicit purpose of generating a SERR#. Before any write to the ERRSTS register, the MEF and SEF bits must be checked, since any write to this register will clear these bits. For example, when software is writing to the ERRSTS register for clearing SEF or MEF bits, SERR# should be disabled. In systems which use SMI# for reporting ECC errors, when SERR# is asserted due to an uncorrectable error, the SMI routine should do the following sequence:<br><br>1.    Read the ECC error flags (MEF, SEF) to determine that an uncorrectable error has occurred.<br><br>2.    Disable SERR# by setting PCICMD [SERRE]='0'.<br><br>3.    Clear MEF bit by writing '1' to it.<br><br>4.    Enable SERR# again.<br><br>5.    Write '1' to MEF bit to create an NMI event. |
| 3:1 | **Single-Bit First Row Error (SBFRE)**  This field is the encoded value of the DRAM row associated with the first single bit error. When an error is detected, this field is updated and the SEF bit is set. This field will be locked (no further updates) until the SEF flag has been reset. When SEF=0, the value in this field is undefined. |

| Bit | Description |
|-----|-------------|
| 0 | **Single-Bit (correctable) ECC Error Flag (SEF)** If this bit is set to 1, the TXC detected and corrected a single bit-error. If SERR# signaling is enabled by the corresponding bit in ERRCMD, and SERRE in PCICMD is set, the error is reported via the SERR# pin.  Software must write a 1 to clear this bit, and unlock the SBFRE field. |

### 3.2.23.    ERRSYN — ERROR SYNDROME REGISTER

Address Offset:        92h
Default:               00000000
Access:                Read Only

This register contains the latest non-zero error syndrome, associated with the error row identified by either the MBFRE or SBFRE fields.

| Bit | Description |
|-----|-------------|
| 7:0 | **Error Syndrome.**  When ECC is enabled, bits 7:0 contain the latest non-zero error syndrome. The error syndrome remains latched until software unlocks the respective field by clearing either the MEF or SEF bit.  Multi-bit errors take priority over single-bit errors, so that if both MEF and SEF are signaled, the syndrome will be the one associated with the multi-bit error (MBFRE).  In parity mode, this contents of this register are undefined.  Writes to ERRSYN have no effect. |

## 4.0.  FUNCTIONAL DESCRIPTION

This section provides a functional description of the TXC including the Host bus interface, PCI bus interface, DRAM interface, L2 cache interface, PCI bus arbitration, and clock generation.

## 4.1.  Host Interface

The Host Interface of the TXC is designed to support the Pentium processor. The host interface of the TXC supports 50 MHz, 60 MHz, and 66 MHz bus speeds. The TXC also supports the Pentium processors's dual processing mode. The TXC supports the Pentium processor with a full 64-bit data bus, 32-bit address bus, and associated internal write-back cache logic. Host bus addresses are decoded by the TXC for accesses to main memory, PCI memory, and PCI I/O. The TXC also supports the pipelined addressing capability of the Pentium processor.

## 4.2.  PCI Interface

The TXC is fully compliant to the PCI 2.1 Specification.  The TXC integrates a high performance interface to the PCI local bus taking full advantage of the high bandwidth and low latency of the PCI. Five PCI masters are supported by the integrated arbiter including four general purpose PCI masters and an optimized arbitration for the PCI-to-ISA bridge. The PCI-to-ISA bridge arbitration dynamically allocates bandwidth to PIIX3 to optimize system latencies for superior Universal Serial Bus (USB) performance.

TXC enhances the basic snoop protocol by allowing pipelining of snoops between two PCI cycles. Thus, more efficient back-to-back frames are possible, especially for short burst cycles.

The PCI interface does not assert PLOCK# on a write cycle, unless PLOCK# was already properly established by a prior read (i.e., read completed without a retry).  If a locked cycle is initiated to a read-only PAM region, the cycle is handled on the host bus as a locked cycle, and the write propagates to PCI as a normal, unlocked, write cycle.

## 4.3.  Secondary Cache Interface

The TXC integrates a high performance write-back second level cache controller using internal/external tags and provides a full first level and second level cache coherency mechanism. The second level cache is direct mapped, non-sectored, and supports a write-back cache policy. Cache lines are allocated on read misses (no write allocate).

The second level cache can be configured to support either a 256-Kbyte or 512-Kbyte cache using synchronous pipelined burst SRAMs. (Note that the TXC does not support asynchronous or burst SRAMs for the cache memory.) For a 256-Kbyte configuration, an 8kx8 standard SRAM is used to store the tags (64-Mbyte memory space cacheability limit). For the 512-Kbyte configuration, a 16kx8 standard SRAM is used to store the tags and the valid bits tags (64-Mbyte memory space cacheability limit). The TXC also supports an optional extended memory space cacheability limit of 512 Mbytes.

A second level cache line is 32 Bytes wide. In the 256-Kbyte configurations, the second level cache contains 8k lines, while the 512-Kbyte configurations contain 16k lines. Valid and modified status bits are kept on a per-line basis. Cacheability of the entire memory space in first level cache is supported. The memory space cacheability for the secondary cache is either 64 Mbytes or 512 Mbytes of main memory, depending on the programming of the ECE bit in the CC register. Table 6 shows the tag sizes needed to support different sizes of cacheability. Only main memory controlled by the TXC DRAM interface is cached. PCI memory is not cached.

**Table 6.  Cacheability**

| Cache Size | Tag Size | Cacheability |
|---|---|---|
| 256 Kbyte | 8K by 8 bits | 64 Mbyte |
| 256 Kbyte | 8K by 9 bits | 128 Mbyte |
| 256 Kbyte | 8K by 10 bits | 256 Mbyte |
| 256 Kbyte | 8K by 11 bits | 512 Mbyte |
| 512 Kbyte | 16K by 8 bits (including valid bit) | 64 Mbyte |
| 512 Kbyte | 16K by 9 bits (including valid bit) | 128 Mbyte |
| 512 Kbyte | 16K by 10 bits (including valid bit) | 256 Mbyte |
| 512 Kbyte | 16K by 11 bits (including valid bit) | 512 Mbyte |

Table 7 shows the different standard SRAM access time requirements for different host clock frequencies.

**Table 7.  SRAM Access Time Requirements**

| Host Clock Frequency (MHz) | Pipelined Burst Clock-to-Output Access Time (ns) | Tag RAM Cycle Time (ns) |
|---|---|---|
| 50 | 13.5 | 20 |
| 60 | 10 | 15 |
| 66 | 8.5 | 15 |

### 4.3.1.    CLOCK LATENCIES

Table 8 lists the latencies for various processor transfers to/from the second level cache.

**Table 8.  Second Level Cache Latencies with Pipelined Burst SRAM**

| Cycle Type | HCLK Count |
|---|---|
| Burst Read | 3-1-1-1 |
| Burst Write (write back) | 3-1-1-1 |
| Single Read | 3 |
| Single Write | 3 |
| Pipelined Back-to-Back Burst Reads | 3-1-1-1, 1-1-1-1 (note 1) |
| Pipelined Back-to-Back Burst Reads (dual bank) | 3-1-1-1, 2-1-1-1 (note 1) |

**NOTES:**

1.    The back-to-back cycles do not account for CPU idle clocks between cycles.

### 4.3.2. SNOOP CYCLES

The snoop (i.e., inquire) cycle is used to probe the first level and second level caches when a PCI master attempts to access main memory. This is done in order to maintain coherency between the first and second level caches and main memory.  Each cache line accessed during PCI master DRAM transfers is snooped by the TXC.

To maintain optimum PCI bandwidth to DRAM, the TXC utilizes a "snoop ahead" algorithm.  Once the snoop for the first cache line of a transfer has completed, the TXC will automatically snoop the next sequential cache line. This snoop ahead policy eliminates most data stall delays while waiting for snoop results.

**Reads**

When a write-back occurs due to a PCI read cycle snoop, the resulting write-back data is returned to the merge buffer.  Data from the merge buffer is then returned directly to PCI, eliminating the delay to perform the write-back to DRAM and a subsequent DRAM read before returning valid data to PCI.

**Writes**

When a write-back occurs due to a PCI write cycle snoop, the resulting write-back data is merged with the incoming PCI write data. The PCI data will not stall while waiting for a write-back to complete unless there is no space in the merge buffer for the write-back data.

### 4.3.3. CACHE ORGANIZATION

Figure 3 and Figure 4 show the connections between the TXC and the external tag RAM and data SRAM.
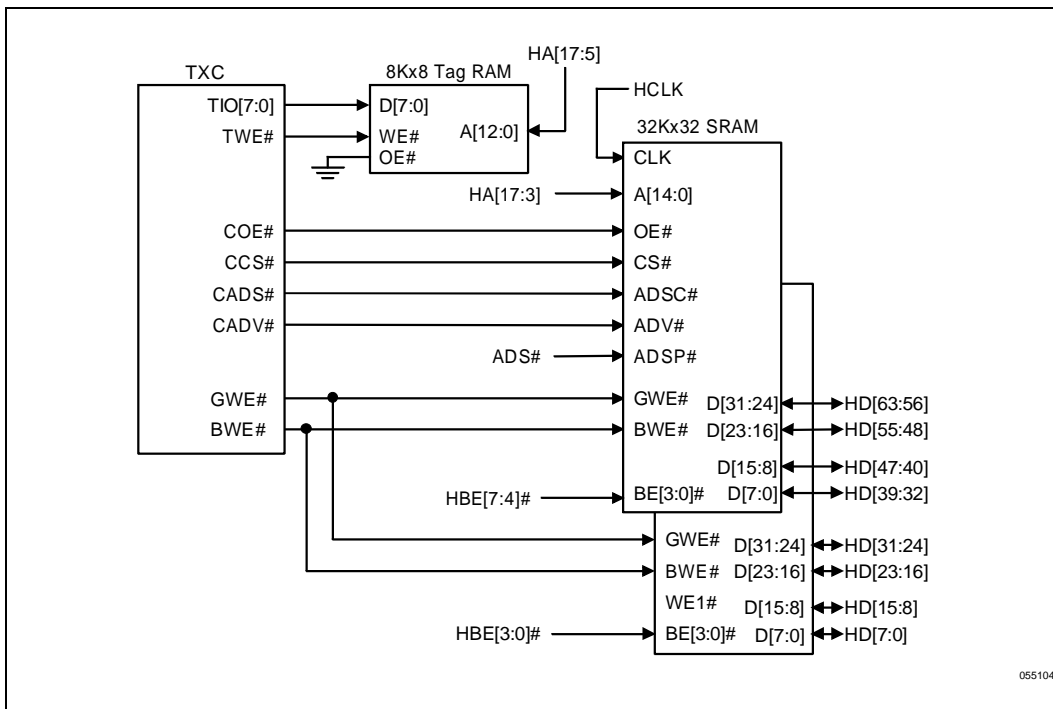


**Figure 3.  256 KB Second Level Cache with PBSRAM (64 MB cacheability)**

![intel] logo

**4.3.4.    DRAM CACHE**

TXC supports DRAM cache. The TIO[10] pin has to be pulled up for DRAM cache.  Extended cache and DRAM cache are mutually exclusive.

For single processor systems using a DRAM cache, register 51, bit 5 in the 82439HX should be set to a 1. For dual processor systems using a DRAM cache and using the cC0 (or later) stepping of the Pentium® processor, this bit should also be set to a 1.  For dual processor systems using earlier steppings of the Pentium® processor, the DRAM cache is not supported.
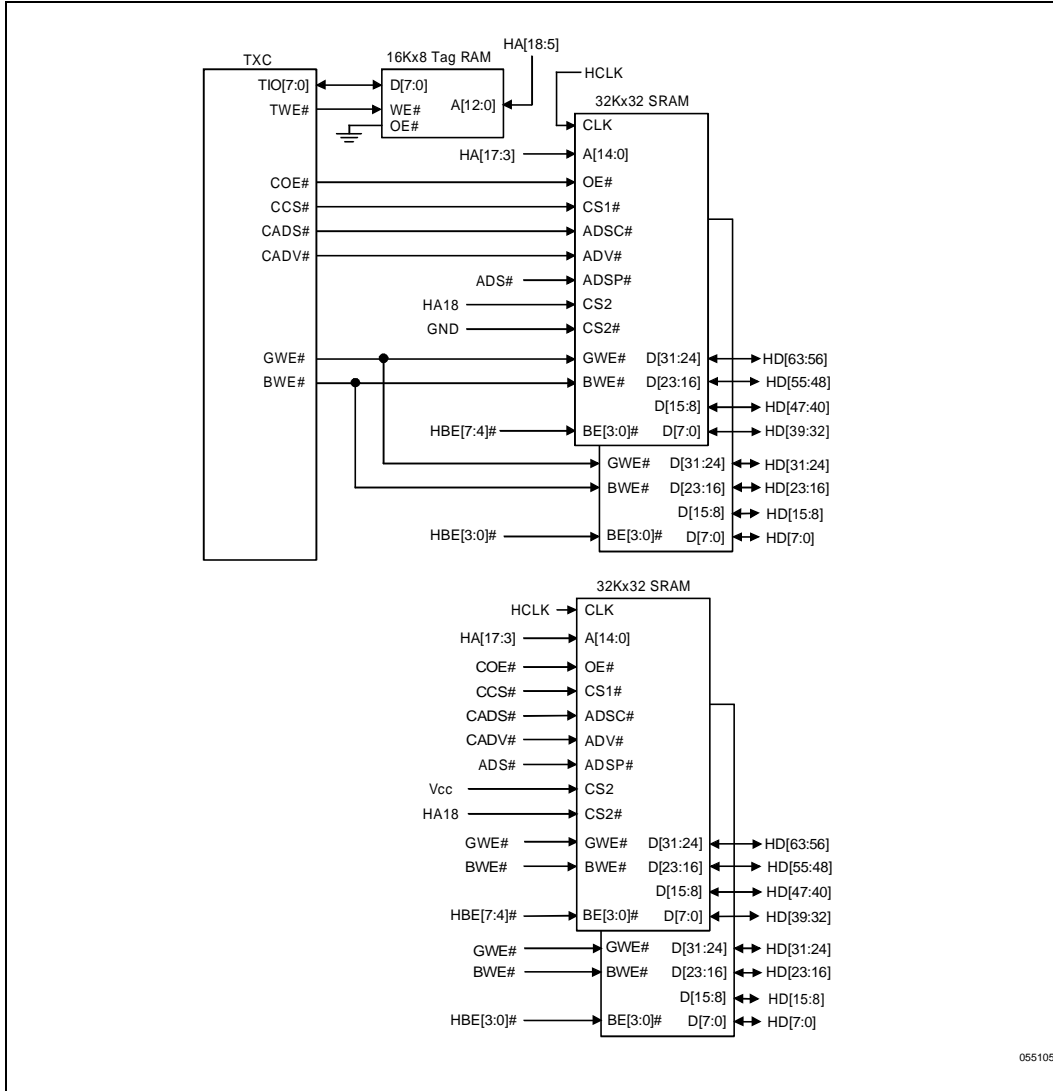


**Figure 4.  512 KB Second Level Cache with PBSRAM (64 MB cacheability)**

## 4.4. DRAM Interface

The TXC integrates a DRAM controller that supports a 64-bit memory array and main memory sizes from 4 Mbytes to 512 Mbytes. The TXC generates all control signals (such as RAS#, CAS#, WE# (using MWE#)) and multiplexed addresses for the DRAM array. The address and data flows through the TXC for all DRAM accesses.

The TXC supports industry standard 32- and 36-bit wide memory modules with fast page-mode DRAMs and supports EDO (Extended Data Out) DRAMs. Twelve multiplexed address lines (MA[11:0]) support 512Kx32/36, 1Mx32/36, 2Mx32/36, 4Mx32/36, 8Mx32/36, and 16Mx32/36 SIMMs, both symmetrical and asymmetrical addressing. Eight RAS# lines support up to eight rows of DRAM. Eight CAS# lines allow byte control over the array during write operations. The TXC supports 50 ns, 60 ns, and 70 ns DRAMs (both single and double-sided SIMM's). The TXC also provides an automatic CBR refresh, at a rate of 1 refresh per 15.6 microseconds at 66 MHz, 60 MHz, and 50 MHz.

The DRAM interface of the TXC is configured by the DRAM Control Mode register, DRAM Extended Control register, DRAM Timing (DRAMT) register, the eight DRAM Row Boundary (DRB) registers, and the DRAM Row Type (DRT) registers. The DRAM Control and Timing registers configure the DRAM interface to select fast page-mode or EDO DRAMs, RAS timings, and CAS rates. The eight DRB registers define the size of each row in the memory array, enabling the TXC to assert the proper RAS# line for accesses to the array.

Seven Programmable Attribute Map (PAM) registers are used to specify the cacheability, PCI enable, and read/write status of the memory space between 640 Kbytes and 1 Mbyte. Each PAM register defines a specific address area enabling the system to selectively mark specific memory ranges as cacheable, read-only, write-only, read/write, or disabled. When a memory range is disabled, all CPU accesses to that range are forwarded to PCI.

The TXC also supports one of two memory holes, either from 512 Kbytes to 640 Kbytes or from 15 Mbytes to 16 Mbytes in main memory. Accesses to the memory holes are forwarded to PCI. The memory hole can be enabled/disabled through the DRAM Control register. All other memory from 1 Mbyte to 512 Mbytes is read/write and is cacheable.

The SMRAM memory space is controlled by the SMRAM Control register. This register selects if the SMRAM space is enabled, opened, closed, or locked.

### 4.4.1. DRAM ORGANIZATION

Figure 5 illustrates a 4-SIMM configuration that supports a motherboard row and 4 double-sided SIMM's. A row in the DRAM array is made up of two SIMM's that share a common RAS# line. RAS0# is routed to motherboard memory and comprises row 0. Within any given row, the two SIMMs must be the same size. Among the rows, SIMM densities can be mixed in any order. Each row is controlled by 8 CAS lines. EDO and Standard page mode DRAM's can be mixed between rows; however, a given row must contain only one type of DRAM. When DRAM types are mixed (EDO and standard page mode) each row will run optimized for that particular type of DRAM.
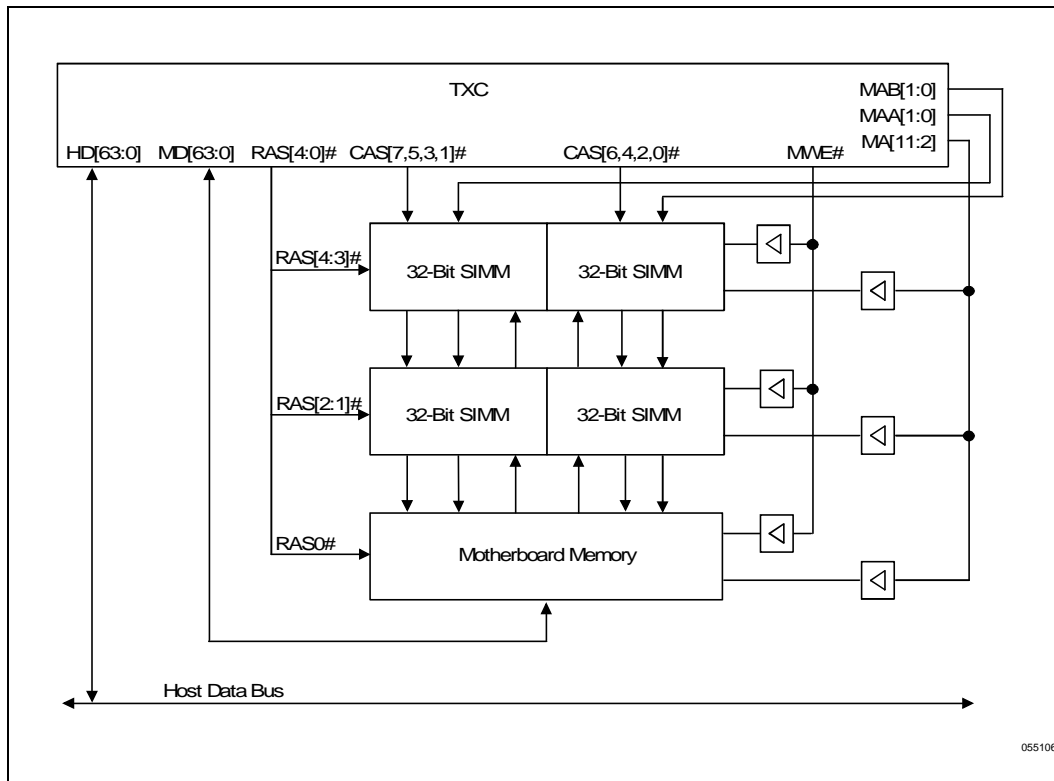
**Figure 5. SIMM Socket**

The fastest possible leadoff and burst times are easier to achieve by using a subset of the RAS# lines. Depending on the system layout, the DRAMEC[MAD] configuration field can be used to provide optimum address drive for a small number of rows.

**Rules for Populating the Memory Array**

The following rules apply to the SIMM configuration.

- SIMM sockets must be populated in pairs. The memory array is 64- or 72-bits wide.

- SIMM sockets can be populated in any order (i.e., memory for RAS0# does not have to be populated before memory for RAS[2:1]# or RAS[4:3]# are used).

- SIMM socket pairs need to be populated with the same densities. For example, SIMM sockets for RAS[2:1]# should be populated with identical densities. However, SIMM sockets for RAS[4:3]# can be populated with different densities than the SIMM socket pair for RAS[2:1]#.

- EDO and standard page mode can both be used. However, only one type should be used per SIMM socket pair. For example, in the table shown below SIMM sockets for RAS[2:1]# can be populated with EDOs while SIMM sockets for RAS[4:3]# can be populated with standard page mode. If different memory is used for different rows, each row will be optimized for that type of memory.

- The DRAM Timing register which provides the DRAM speed grade control for the entire memory array must be programmed to use the timings of the slowest DRAMs installed.

39

**Table 9.  Sample Of Possible Mix And Match Options For SIMMs**

| RASx# Lines | | | | | DRB Registers | | | | | | | | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1,2 | 3,4 | 5,6 | 7 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Mem |
| 8M/S | 0 | 0 | 0 | 0 | 02h | 02h | 02h | 02h | 02h | 02h | 02h | 02h | 8 MB |
| 0 | 4M/S | 0 | 0 | 0 | 0 | 01h | 01h | 01h | 01h | 01h | 01h | 01h | 4 MB |
| 0 | 0 | 8M/S | 0 | 0 | 0 | 0 | 0 | 02h | 02h | 02h | 02h | 02h | 8 MB |
| 8M/S | 8M/S | 0 | 0 | 0 | 02h | 04h | 04h | 04h | 04h | 04h | 04h | 04h | 16 MB |
| 4M/S | 8M/S | 8M/S | 0 | 0 | 01h | 03h | 03h | 05h | 05h | 05h | 05h | 05h | 20 MB |
| 8M/S | 16M/S | 0 | 0 | 0 | 02h | 06h | 06h | 06h | 06h | 06h | 06h | 06h | 24 MB |
| 8M/S | 16M/S | 16M/S | 0 | 0 | 02h | 06h | 06h | 0Ah | 0Ah | 0Ah | 0Ah | 0Ah | 40 MB |
| 8M/S | 32M/S | 0 | 0 | 0 | 02h | 0Ah | 0Ah | 0Ah | 0Ah | 0Ah | 0Ah | 0Ah | 40 MB |
| 8M/S | 32M/S | 32M/S | 0 | 0 | 02h | 0Ah | 0Ah | 12h | 12h | 12h | 12h | 12h | 72 MB |
| 0 | 0 | 32M/S | 0 | 0 | 0 | 0 | 0 | 08h | 08h | 08h | 08h | 08h | 32 MB |
| 4 M/S | 32M/D | 64M/D | 0 | 0 | 01h | 05h | 09h | 11h | 19h | 19h | 19h | 19h | 104 MB |
| 0 | 64M/D | 64M/D | 0 | 0 | 0 | 08h | 10h | 18h | 20h | 20h | 20h | 20h | 128 MB |
| 8 M/S | 64M/D | 64M/D | 0 | 0 | 02h | 0Ah | 12h | 1Ah | 22h | 22h | 22h | 22h | 136 MB |
| 0 | 0 | 64M/D | 0 | 0 | 0 | 0 | 0 | 08h | 10h | 10h | 10h | 10h | 64 MB |

| RASx# Lines | | | | DRB Registers | | | | | | | | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0,1 | 2,3 | 4,5 | 6,7 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Mem |
| 64M/D | 64M/D | 64M/D | 64M/D | 08h | 10h | 18h | 20h | 28h | 30h | 38h | 40h | 256 MB |
| 64M/S | 64M/S | 64M/S | 64M/S | 10h | 10h | 20h | 20h | 30h | 30h | 40h | 40h | 256 MB |
| 128M/D | 128M/D | 128M/D | 128M/D | 10h | 20h | 30h | 40h | 50h | 60h | 70h | 80h | 512 MB |
| 128M/S | 128M/S | 128M/S | 128M/S | 20h | 20h | 40h | 40h | 60h | 60h | 80h | 80h | 512 MB |

**NOTES:**

"S" denotes single-sided SIMM's, "D" denotes double-sided SIMM's.

Table 10 provides a summary of the characteristics of memory configurations supported by the TXC. Minimum values listed are obtained with single-sided SIMMs and maximum values are obtained with double-sided SIMMs. Note that for a 64-bit wide memory array, a minimum of two 32-/36-bit wide DRAM SIMM configuration is required. The minimum values used are also the smallest upgradeable memory size.

intel®

**Table 10. Minimum (Upgradeable) and Maximum Memory Size for each configuration**

| DRAM Tech. | DRAM Density | DRAM Width | DRAM SIMM SS x32 or 36 | DRAM SIMM DS x32 or 36 | DRAM Addressing | Row | Column | Min (UP) | Max |
|---|---|---|---|---|---|---|---|---|---|
| 4M | | | | | | | | | |
| | 512K | 8 | 512K | 1M | Asymmetric | 10 | 9 | 4 MB | 32 MB |
| | 1M | 4 | 1M | 2M | Symmetric | 10 | 10 | 8 MB | 64 MB |
| 16M | | | | | | | | | |
| | 1M | 16 | 1M | 2M | Symmetric | 10 | 10 | 8 MB | 64 MB |
| | 2M | 8 | 2M | 4M | Asymmetric | 11 | 10 | 16 MB | 128 MB |
| | 4M | 4 | 4M | 8M | Symmetric | 11 | 11 | 32 MB | 256 MB |
| | 4M | 4 | 4M | 8M | Asymmetric | 12 | 10 | 32 MB | 256 MB |
| 64M | | | | | | | | | |
| | 4M | 16 | 4M | 8M | Symmetric | 11 | 11 | 32 MB | 256 MB |
| | 8M | 8 | 8M | 16M | Asymmetric | 12 | 11 | 64 MB | 512 MB |
| | 16M | 4 | 16M | 32M | Symmetric | 12 | 12 | 128 MB | 512 MB |

The memory organization (Figure 6) represents the maximum 512 Mbytes of address space. Accesses to memory space above top of DRAM, video buffer, or the memory gaps (if enabled) are forwarded to PCI, and these regions are not cacheable. Below 1 Mbyte, there are several memory segments that have selectable cacheability. None of the DRAM space occupied by the video buffer or the memory space gaps is remapped and is therefore "lost".
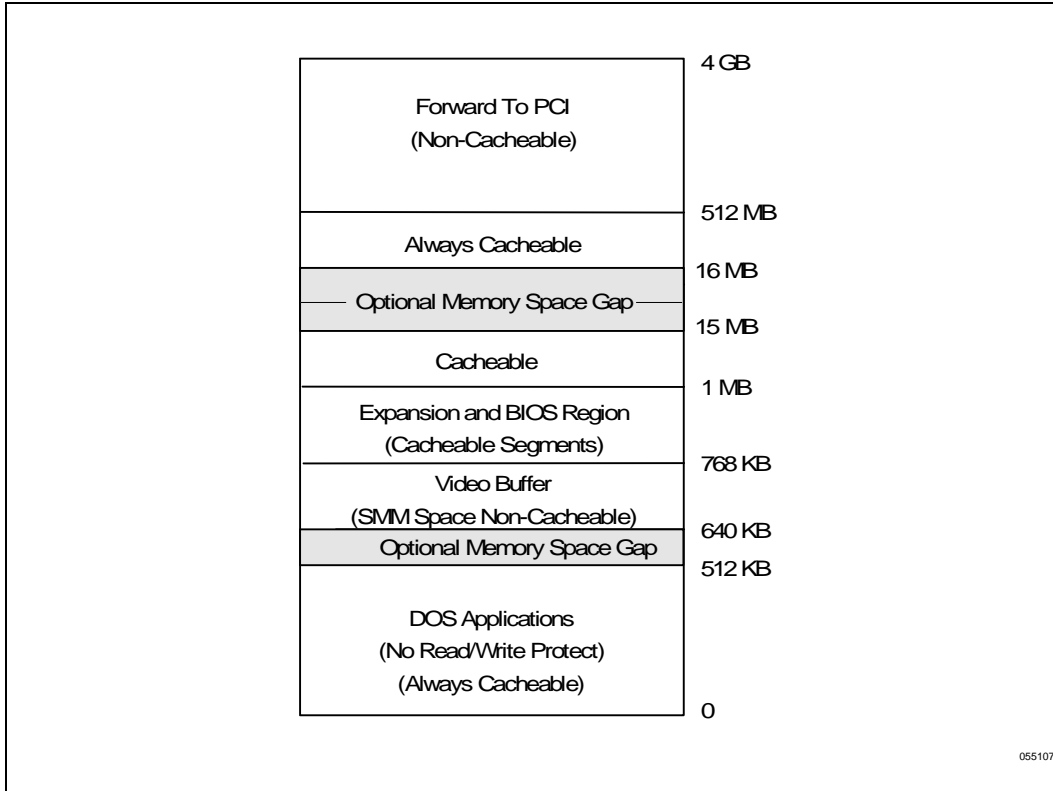
**Figure 6. Memory Space Organization**

### 4.4.2. DRAM ADDRESS TRANSLATION

The multiplexed row/column address to the DRAM memory array is provided by the MA[11:0] signals. The MA[11:0] bits are derived from the host or PCI address bus as defined by Table 11. Note that the lower 10 row address bits and all the column address bits are fixed. The upper 2 row address bit selection varies for each row, and is dependent on the row mux option programmed for that row.

**Table 11. DRAM Address Translation**

| Row Mux Option | MA [11:0] | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 64M | Row | A25 | Default | | | | | | | | | | |
| Default | Row | A24 | A23 | A21 | A20 | A19 | A18 | A17 | A16 | A15 | A14 | A13 | A12 |
| | Col | A26 | A24 | A22 | A11 | A10 | A9 | A8 | A7 | A6 | A5 | A4 | A3 |

The DRAM addressing and the size supported by these options is shown in Table 12.

Note: 12 x 8 Asymmetric DRAM is not supported.

**Table 12.  Memory Mapping Options**

| Memory Organization | Addressing | Address size |
|---|---|---|
| **4 MB** | | |
| 512K x8 | Asymmetric | 10 x 9 |
| 1M x 4 | Symmetric | 10 x 10 |
| **16 MB** | | |
| 1M x 16 | Symmetric | 10 x 10 |
| 2M x 8 | Asymmetric | 11 x 10 |
| 4M x 4 | Symmetric | 11 x 11 |
| 4M x 4 | Asymmetric | 12 x 10 |
| **64 MB** | | |
| 4M x 16 | Symmetric | 11 x 11 |
| 8M x 8 | Asymmetric | 12 x 11 |
| 16M x 4 | Symmetric | 12 x 12 |

#### 4.4.3.    DRAM TYPES

**FPM Mode**

The TXC, as a default, supports the standard fast page mode (FPM) DRAM.

**EDO Mode**

Extended Data Out DRAM is designed to improve the DRAM read performance. EDO DRAM holds the memory data valid until the next CAS# falling edge. Compared to standard page mode DRAM which tri-states the memory data when CAS# negates to precharge. With EDO the CAS# precharge overlaps the memory data valid time. This allows CAS# to negate earlier while still satisfying the memory data valid window time.

**Auto-Detection**

A special timing mode (DRAMC[EDME]) may be used for BIOS to detect EDO DRAM's on a bank-by-bank basis. The BIOS first assumes all memory is EDO (sets EDO) and writes a pattern sequence to the main memory. The BIOS then sets the EDO/FPM detect bit to enable the TXC's hardware detect mode. The BIOS then reads the memory location that was previously written. Any read subsequent to the setting of the EDO/FPM detect mode bit will go through the following sequence:

- The TXC waits for the refresh timeout and then asserts CAS# for 3 host clocks.

- The TXC waits for another refresh timeout.

- For a timeout, the DRAM controller latches the data present on the MDxx bus currently and then drives the data onto the host data bus.

The BIOS reads the latched data to determine the type of DRAM present. If the DRAM was FPM type, the negation of the CASx# should have tri-stated the DRAM buffers and, due to the internal pulldowns on the MDxx lines, the data read by the BIOS would be all zeros. If the DRAM was EDO type, the negation of the CASx# would have kept the DRAM buffers active and the BIOS would have read the written pattern, since the data is still driven. Any subsequent DRAM read request will start this sequence again, so all writes must be done before any reads. All writes to the DRAM are treated normally, even if the auto-detection mode is enabled.

Once the type of DRAM has been detected, this information must then be programmed into the DRAM Row Type register. The TXC uses the DRAM Row Type information in conjunction with the DRAM timings set in the DRAM Timing register to configure DRAM accesses optimally.

### 4.4.4. DRAM PERFORMANCE

The DRAM performance is controlled by the DRAM timing, processor pipelining, and by the type of DRAM used (EDO or standard page mode).

Tables 12 and 13 depict DRAM mode optimum timings for a 64-bit wide double-sided row of EDO SIMMs (i.e., two RAS# lines), without parity and ECC enabled. Expected performance differs depending on DRAM type (EDO vs. FPM), parity or ECC enabled/disabled, DRAM access time, and bus frequency.

**Read Cycles**

DRAM read performance is measured in two parts. The first part (leadoff) consists of data read from DRAM and is affected by the DRAM technology's page-dependent access time, level of pipelining, and the TXC's ability in identifying cache hits or misses. The second part (burst) is determined by the DRAM technology's ability to put out data bursts.

**Write Cycles**

DRAM write performance is measured in two parts. The first part is when data is posted into the TXC, and thus, has a leadoff dependent on the availability of the DRAM write posting buffers space. The second part is the latency to write the data from the TXC to memory.

**Table 13.  CPU to DRAM Read Performance Summary (Notes 2 and 5, Table 13)**

| DRAM Speed | DRAM Type | Performance (in host clocks) | |
|---|---|---|---|
| | | **50/60 MHz** | **66 MHz** |
| **Read (Burst Rate)** | | | |
| 50 ns | EDO | x-222 | x-222 |
| | FPM | x-333 | x-333 |
| | | | |
| 60 ns | EDO | x-222 | x-222 |
| | FPM | x-333 | x-333 |
| | | | |
| 70 ns | EDO | x-333 | x-333 |
| | FPM (lightly loaded) | x-333 | x-444 |
| **Read Page Hit/Page Miss/Row Miss Leadoff Rate** | | | |
| Page Hit | | | |
| 60 ns | EDO/FPM | 4 | 5 |
| Row Miss (Note 1) | | | |
| 60 ns | EDO/FPM | 7 | 8 |
| Page Miss | | | |
| 60 ns | EDO/FPM | 10 | 11 |
| **Back-to-Back Burst Reads Page Hit** | | | |
| 60 ns | EDO | 4-2223222 | 5-2223222 |
| 60 ns | FPM | 4-3333333 | 5-3333333 |

**Table 14.  CPU to DRAM  Write Performance Summary (Note 5)**

| DRAM Speed | DRAM Type | Performance (in host clocks) | |
|---|---|---|---|
| | | **50/60 MHz** | **66 MHz** |
| **Posted Single Write (Note 3,4)** | | | |
| 60 ns | EDO/FPM | | 3-3-3 |
| **Posted Burst Write (Note 3,4)** | | | |
| 60 ns | EDO/FPM | | 3-111 |
| **Single Retire Hit** | | | |
| 60 ns | FPM | 2 | 3 |
| | EDO | 2 | 2 |
| **Single Retire Row Miss (Note 6)** | | | |
| 60 ns | FPM | 4 | 5 |
| | EDO | 4 | 4 |
| **Single Retire Page Miss (Note 6)** | | | |
| 60 ns | FPM | 7 | 8 |
| | EDO | 7 | 7 |
| **Retire Burst** | | | |
| 60 ns | FPM | - 222 | -333 |
| | EDO | -222 | -222 |

**NOTES:**

1.  The above row miss cycles assume that the new page is closed from the prior cycle. Due to the MA[11:2] to RAS# setup requirements, if the page is open, 2 clocks are added to the leadoff.

2.  50/60 MHz allows both Speculative Leadoff  and Turbo Read Leadoff to be enabled, whereas 66 MHz only allows Speculative Leadoff to be enabled.

3.  This cycle timing assumes the write buffer is empty.

4.  Timings are the same for cached or cacheless systems.

5.  Timings with ECC enabled (EDE=1) are adjusted by +1 HCLK for all leadoff latencies. Read Turbo Leadoff enabled is illegal if ECC is enabled or L2 is enabled. Reads don't pass writes except for dirty replacements.

6.  Assumes fast RAS to CAS timings.


**4.4.5.    DRAM REFRESH**

The TXC supports only CAS-before-RAS# (CBR) refresh. The refresh rate is controlled via the DRAM Refresh Rate field in the DRAM Control register. Refresh is only performed on rows that are populated (i.e., "smart refresh").  The controller determines which rows are populated by looking at the DRB registers.

BIOS should turn off refresh during the DRAM sizing. A delay of at least 60 microseconds should also be added once refresh is turned off and before the first write occurs.

**4.4.6.    SYSTEM MANAGEMENT RAM**

SMRAM is placed at A0000-BFFFFh via the SMRAM Space register.

**4.4.7.    DATA INTEGRITY SUPPORT**

ECC or parity can be checked on the DRAM interface. The default status is  parity selected. The DRAM must be populated with 72-bit wide memory to implement ECC or parity.

**4.4.7.1.    Parity**

When parity is enabled, the DRAM parity protection is 8-bit based even parity. If the DRAM array is populated with 64-bit memory (vs 72-bit) the parity logic still logs parity errors in the ERRSTS register. For such DRAM configurations, bit 1 of the ERRCMD register must be 0 (default) to prevent these errors from being signaled via the SERR# mechanism.

BIOS must be aware of the implication of the optional parity support. All physically present DRAM must be first written before SERR#-based NMI generation in the PIIX3 is enabled.

**Detection of 64- versus 72-bit Wide SIMMs.** The TXC only supports parity or ECC properly if all DRAMs are 72-bit wide. A system with a mixture of 64- and 72-bit wide memory should disable parity and ECC. BIOS can detect the 64-bit wide DRAMs (so that it can disable SERR# on parity error) by writing data that forces the parity bits to be all 1's for address A, and then writing data that forces the parity bits to be all 0's in another location, address B.  Now, if address A is read, no parity error will result only if there's a 72-bit wide DRAM present, whereas parity errors would get flagged in the ERRSTS register for a 64-bit wide DRAM. To guard against the chance that somehow the parity pins are being pulled high inadvertently, the test should be redone, this time writing data that forces the parity bits to be all 0's for address A, and all 1's for address B.

**4.4.7.2.    Error Detection and correction**

ECC is an optional data integrity feature provided by the TXC.  The feature provides single-error correction, double-error detection, and detection of all errors confined to a single nibble (SEC-DED-S4ED) for the DRAM memory subsystem. Additional features are provided that enable software-based system management capabilities.

In systems that support ECC, the read burst timings in the DRAMT register must be programmed to x-3-3-3 (or x-4-4-4). For ECC systems, the 32 clock retry feature should be disabled. The feature can be disabled by setting bit 2 at register offset 4FH to 1.

**ECC Generation.** When enabled, the TXC generates an 8-bit protection code for 64-bit data during DRAM write operations. If the original write is less than 64-bits, a read-modify-write operation is performed.

**ECC Checking and Correction.** When enabled, the TXC detects all single and dual-bit errors, and corrects all single-bit errors during DRAM reads. The corrected data is transferred to the requester (CPU or PCI). Note that the corrected data is *not* written back to DRAM.

**Error Reporting.** When ECC is enabled and ERRCMD is used to set SERR# functionality, ECC errors are signaled to the system via the SERR# pin. The TXC can be programmed to signal SERR# on uncorrectable errors, correctable errors, or both. The type of error condition is latched until cleared by software (regardless of SERR# signaling).

When a single or multi-bit error is detected, the offending DRAM row ID is latched in the ERRSTS register in the TXC. The latched value is held until software explicitly clears the error status flag.

**Software Requirements**

**Initialization.** If the ECC feature or parity is enabled, BIOS must take care to properly initialize the memory before enabling the checking. Software should first ensure PCICMD[SERRE] =0, then enable ECC or parity via ERRCMD. Next, the entire DRAM array should be written to ensure valid syndrome/parity bits. Finally, the desired ECC/parity error reporting should be enabled via the ERRCMD and PCICMD registers.

**Parity Error Handling.** Parity error handling should be via the system's normal NMI routines.

**ECC Support Levels.** The TXC allows for various levels of ECC support, depending on the specific platform requirements. The software architecture requirements vary based on the level of support implemented. The levels and basic software implications are summarized in the Table 15.

**Table 15.  ECC Software Levels**

| Level | Features | Software Requirements |
|---|---|---|
| 1 | Error Checking/Correction | Configuration BIOS |
| 2 | Error Checking/Correction<br>Error Scrubbing | Configuration BIOS<br>SMI Scrubbing Routine |
| 3 | Error Checking/Correction<br>Error Scrubbing<br>System Management (e.g. error logging, error isolation, memory remapping, etc.) | Configuration BIOS<br>SMI Scrubbing Routine<br>OS-dependent System management handler/applet |

**Level 1**

Level 1 defines a minimal support level for ECC handling that would use the system's standard NMI routine. The configuration BIOS enables SERR# generation only for uncorrectable errors, and disables SERR# generation for correctable errors. The NMI routine will interpret the uncorrectable error event as a parity error, and typically reboot the system. The SERR#-duration should be initialized to generate a pulsed SERR# event (ERRCMD[SED] = 0).

**Level 2**

Level 2 adds support for error scrubbing of correctable errors using operating system (OS) independent mechanisms. SMI is the preferred mechanism to implement OS-independence. In this case, the SMI handler would be invoked for both correctable and uncorrectable errors. The SERR# output will not be connected to the PCI bus, but will instead connect to external logic that differentiates between SMI and NMI requests. The SERR#-output type should be configured as a normal output (PCON[SOT]=1), and the duration should be initialized to the level mode (ERRCMD[SED] = 1).

For correctable errors, the handler first clears the error flags and then starts a scrub process. The time spent in an SMI routine should be minimized, since interrupts are disabled and OS services (e.g. real time clocks) could be adversely affected. To minimize time spent during the SMI handler, the scrub process should be distributed into small time slices. The handler can setup future SMI events to re-occur based on a hardware timer (e.g. the "Fast Off" green timer in Intel PCIset standard expansion bridges [PIIX3]) until the memory scrub has completed.

The following example estimates the worst-case scrub duration for a single correctable error.

Example Assumptions:

- 128 Mbytes/Row (64 Mbit technology) =4M lines at 32  bytes/line

- Scrub operation memory bound by linefill + write-back (with medium DRAM timings) = 30 clks/line

- SMI scrub time slice budget = 1 msec/SMI

- Fast-off SMI interrupt interval = 100 msec/interrupt

With the above assumptions The scrub time is:

- the time to scrub one row is 4M lines/row  X 30 clks/line X 15 ns/clk = 1.8 msec/row.

- Thus, spreading the scrub time requires: (1.8 sec/row)/(1 msec /SMI) = 1800 SMI events.

- The total duration for the scrub SMI events will be 1800 SMI events * 100 msec/interrupt  = 180 sec.

For uncorrectable errors, the SMI handler should pass the error to the system's normal NMI handler, making it appear as a standard parity error to the software.  To pass the NMI event, the SMI routine should first clear out the ECC error flags (MEF, SEF).  The handler should then generate a pulsed SERR# event by writing a 1 to the MEF flag.  This generates a 1 PCLK SERR# assertion, which will be converted into an NMI interrupt by the expansion bridge.

**Level 3**

Level 3 adds more sophisticated system management functions beyond simple error scrubbing. Typical functions could include error event logging, error isolation, memory remapping, system diagnostics and user interface application. Software to implement Level 3 functions are OS-dependent, and beyond the scope of this document. However, the features used to implement Level 2 functions can also be leveraged in Level 3 systems.

## 4.5.    PCI Bus Arbitration

The TXC's PCI Bus arbiter allows PCI peer-to-peer traffic concurrent with CPU main memory/second level cache cycles. The arbiter supports five PCI masters (Figure 7). REQ[3:0]#/GNT[3:0]# are used by PCI masters other than the PCI-to-ISA expansion bridge (PIIX3). PHLD#/PHLDA# are the arbitration request/grant signals for the PIIX3 and provide guaranteed access time capability for ISA masters. PHLD#/PHLDA# also optimize system performance based on PIIX3 known policies. The arbitration mechanism is enhanced with a Multi-Transaction Timer mechanism. This feature improves the PCI bandwidth allocation to short bursts, an important consideration for example, with typical video capture devices.

**Multi-Transaction Timer (MTT)**

The priority chain algorithm has been enhanced by the Multi-Transaction Timer (MTT), register 70h. The effect of the MTT is to guarantee a minimum time slice on PCI to an agent that keeps its request asserted. (Note that this mechanism differs from the MLT operation, that enforces a maximum time slice for an agent.) The MTT algorithm ensures a fairer bandwidth allocation for PCI devices that generate short burst traffic, or for multi-function devices with several bus master agents behind one physical PCI interface.

**Passive Release and Bus Lock**

To comply with PCI 2.1 latency requirements, the TXC supports passive release. The TXC only supports bus lock mode. The bus lock mode precludes 3rd party locks.

**Enabling Delayed Transactions**

To enable Delayed Transactions, Delayed Transactions (PIIX3 Function 0, 82h[0]), North Bridge Retry Enable (PIIX3 Function 0, 6Ah[7]), and TXC 4Fh[7]), bits should all be set to '1'. When enabling, PIIX3 bits should be set first and then the TXC 4Fh[7]. When disabling, the reverse order must be followed.
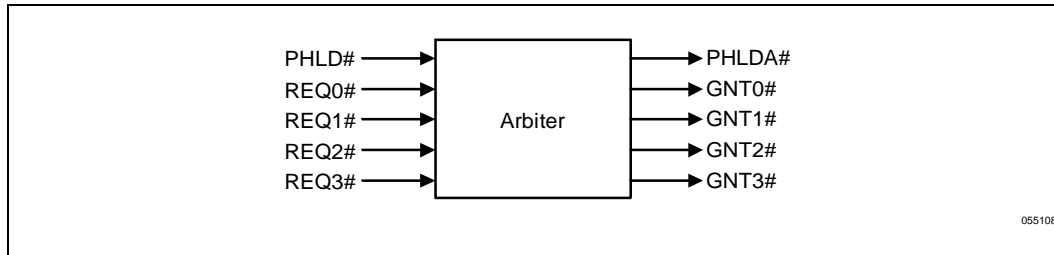
**Figure 7.  PCI Bus Arbiter**

#### 4.5.1.    CPU POLICIES

The CPU never explicitly requests the bus. Instead, the arbiter grants the bus to the CPU when:

- the CPU is the highest priority

- PCI agents do not require main memory (peer-to-peer transfers or bus idle) and the PCI Bus is not currently locked by a PCI master

When the CPU is granted as highest priority, the MLT timer is used to guarantee a minimum amount of system resources to the CPU before another requesting PCI agent is granted. An AHOLD mechanism controls granting the bus to the CPU.

### 4.6.    Clock Generation and Distribution

The skew between any two HCLK loads must be less than or equal to 1 ns. The skew between HCLK at the pin of the TXC and the pin of the CPU must be less than or equal to 0.3 ns. The skew between any two PCLK loads must be less than or equal to 2 ns, the requirement of the PCI Rev 2.0 specification. For HCLK to PCLK ratios of  2-to-1, the rising edge of PCLKIN must be within 1 to 6.0 ns of the rising edge of HCLKIN, with HCLK leading PCLK.

#### 4.6.1.    RESET SEQUENCING

The TXC is asynchronously reset by the PCI reset RST#.  The TXC will change MWE# to its default value (inactive) synchronous to HCLK after RST# is negated.

**Arbiter (Central Resource) Functions on Reset**

The TXC arbiter includes support for PCI central resource functions.  These functions include driving the AD[31:0], C/BE[3:0]#, and PAR signals whenever no one is granted the PCI bus and the bus is idle.  The TXC drives 0's on these signals during these times, plus during RESET.

## 5.0. PINOUT AND PACKAGE INFORMATION

### 5.1. TXC Pinout Information

Figure 8 shows the pin assignments for the TXC.



**Figure 8. TXC Pinout**
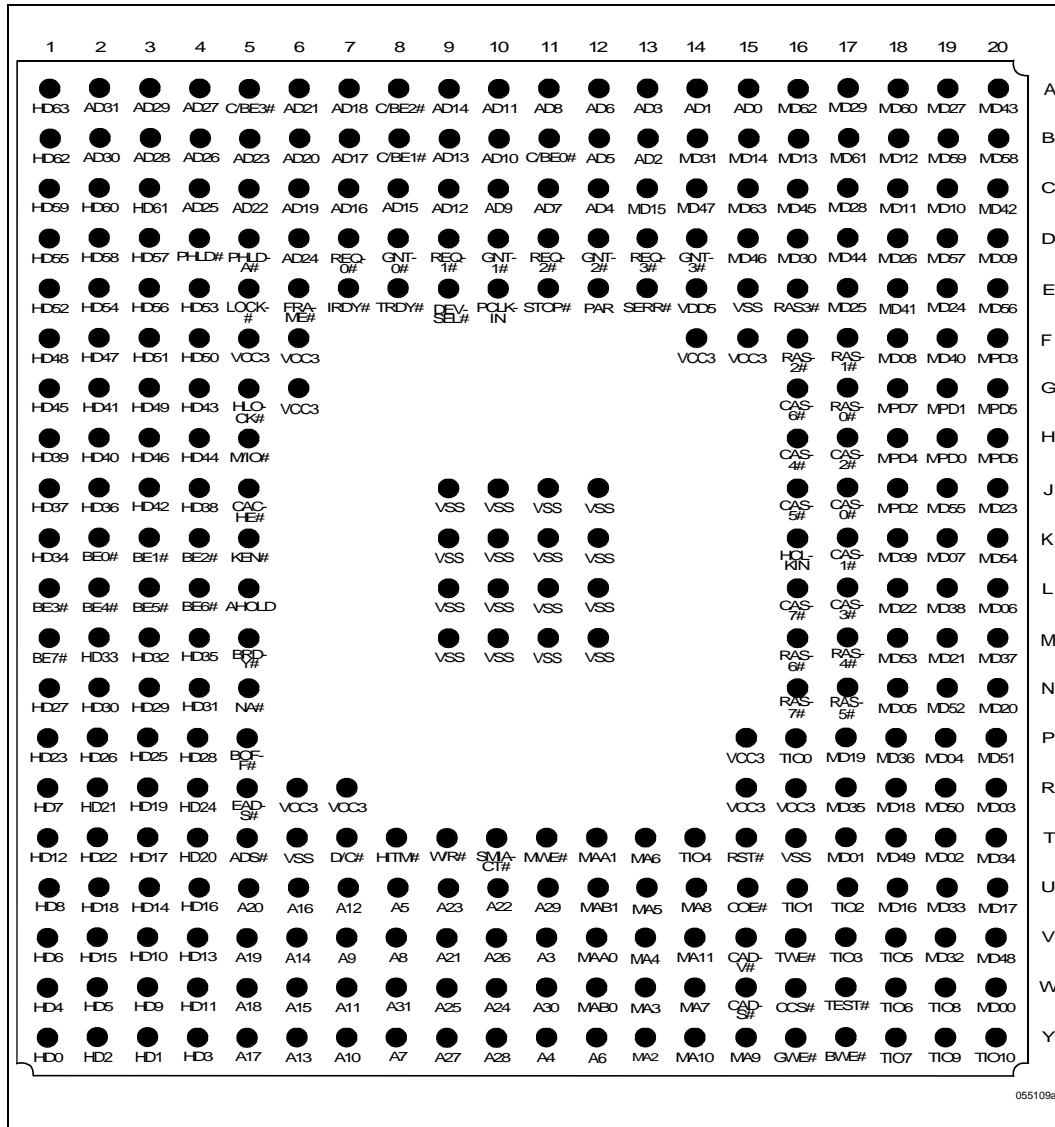
**Table 16.  TXC Alphabetical Pin Assignment**

| Name | Ball # | Type | Name | Ball # | Type | Name | Ball # | Type |
|------|--------|------|------|--------|------|------|--------|------|
| A3 | V11 | I/O | AD1 | A14 | I/O | AD31 | A2 | I/O |
| A4 | Y11 | I/O | AD2 | B13 | I/O | ADS# | T5 | I |
| A5 | U8 | I/O | AD3 | A13 | I/O | AHOLD | L5 | O |
| A6 | Y12 | I/O | AD4 | C12 | I/O | BE0# | K2 | I |
| A7 | Y8 | I/O | AD5 | B12 | I/O | BE1# | K3 | I |
| A8 | V8 | I/O | AD6 | A12 | I/O | BE2# | K4 | I |
| A9 | V7 | I/O | AD7 | C11 | I/O | BE3# | L1 | I |
| A10 | Y7 | I/O | AD8 | A11 | I/O | BE4# | L2 | I |
| A11 | W7 | I/O | AD9 | C10 | I/O | BE5# | L3 | I |
| A12 | U7 | I/O | AD10 | B10 | I/O | BE6# | L4 | I |
| A13 | Y6 | I/O | AD11 | A10 | I/O | BE7# | M1 | I |
| A14 | V6 | I/O | AD12 | C9 | I/O | BOFF# | P5 | O |
| A15 | W6 | I/O | AD13 | B9 | I/O | BRDY# | M5 | O |
| A16 | U6 | I/O | AD14 | A9 | I/O | BWE# | Y17 | O |
| A17 | Y5 | I/O | AD15 | C8 | I/O | C/BE0# | B11 | I/O |
| A18 | W5 | I/O | AD16 | C7 | I/O | C/BE1# | B8 | I/O |
| A19 | V5 | I/O | AD17 | B7 | I/O | C/BE2# | A8 | I/O |
| A20 | U5 | I/O | AD18 | A7 | I/O | C/BE3# | A5 | I/O |
| A21 | V9 | I/O | AD19 | C6 | I/O | CACHE# | J 5 | O |
| A22 | U10 | I/O | AD20 | B6 | I/O | CADS# | W15 | O |
| A23 | U9 | I/O | AD21 | A6 | I/O | CADV# | V15 | O |
| A24 | W10 | I/O | AD22 | C5 | I/O | CAS0# | J 17 | O |
| A25 | W9 | I/O | AD23 | B5 | I/O | CAS1# | K17 | O |
| A26 | V10 | I/O | AD24 | D6 | I/O | CAS2# | H17 | O |
| A27 | Y9 | I/O | AD25 | C4 | I/O | CAS3# | L17 | O |
| A28 | Y10 | I/O | AD26 | B4 | I/O | CAS4# | H16 | O |
| A29 | U11 | I/O | AD27 | A4 | I/O | CAS5# | J16 | O |
| A30 | W11 | I/O | AD28 | B3 | I/O | CAS6# | G16 | O |
| A31 | W8 | I/O | AD29 | A3 | I/O | CAS7# | L16 | O |
| AD0 | A15 | I/O | AD30 | B2 | I/O | CCS# | W16 | O |

| Name | Ball # | Type | Name | Ball # | Type | Name | Ball # | Type |
|------|--------|------|------|--------|------|------|--------|------|
| COE# | U15 | O | HD20 | T4 | I/O | HD51 | F3 | I/O |
| D/C# | T7 | I | HD21 | R2 | I/O | HD52 | E1 | I/O |
| DEVSEL# | E9 | I/O | HD22 | T2 | I/O | HD53 | E4 | I/O |
| EADS# | R5 | O | HD23 | P1 | I/O | HD54 | E2 | I/O |
| FRAME# | E6 | I/O | HD24 | R4 | I/O | HD55 | D1 | I/O |
| GNT0# | D8 | I | HD25 | P3 | I/O | HD56 | E3 | I/O |
| GNT1# | D10 | I | HD26 | P2 | I/O | HD57 | D3 | I/O |
| GNT2# | D12 | I | HD27 | N1 | I/O | HD58 | D2 | I/O |
| GNT3# | D14 | I | HD28 | P4 | I/O | HD59 | C1 | I/O |
| GWE# | Y16 | O | HD29 | N3 | I/O | HD60 | C2 | I/O |
| HCLKIN | K | I | HD30 | N2 | I/O | HD61 | C3 | I/O |
| HD0 | Y1 | I/O | HD31 | N4 | I/O | HD62 | B1 | I/O |
| HD1 | Y3 | I/O | HD32 | M3 | I/O | HD63 | A1 | I/O |
| HD2 | Y2 | I/O | HD33 | M2 | I/O | HITM# | T8 | I |
| HD3 | Y4 | I/O | HD34 | K1 | I/O | HLOCK# | G5 | I |
| HD4 | W1 | I/O | HD35 | M4 | I/O | IRDY# | E7 | I/O |
| HD5 | W2 | I/O | HD36 | J 2 | I/O | KEN#/INV | K5 | O |
| HD6 | V1 | I/O | HD37 | J1 | I/O | LOCK# | E5 | I/O |
| HD7 | R1 | I/O | HD38 | J 4 | I/O | M/IO# | H5 | I |
| HD8 | U1 | I/O | HD39 | H1 | I/O | MA2 | Y13 | O |
| HD9 | W3 | I/O | HD40 | H2 | I/O | MA3 | W13 | O |
| HD10 | V3 | I/O | HD41 | G2 | I/O | MA4 | V13 | O |
| HD11 | W4 | I/O | HD42 | J 3 | I/O | MA5 | U13 | O |
| HD12 | T1 | I/O | HD43 | G4 | I/O | MA6 | T13 | O |
| HD13 | V4 | I/O | HD44 | H4 | I/O | MA7 | W14 | O |
| HD14 | U3 | I/O | HD45 | G1 | I/O | MA8 | U14 | O |
| HD15 | V2 | I/O | HD46 | H3 | I/O | MA9 | Y15 | O |
| HD16 | U4 | I/O | HD47 | F2 | I/O | MA10 | Y14 | O |
| HD17 | T3 | I/O | HD48 | F1 | I/O | MA11 | V14 | O |
| HD18 | U2 | I/O | HD49 | G3 | I/O | MAA0 | V12 | O |
| HD19 | R3 | I/O | HD50 | F4 | I/O | MAA1 | T12 | O |

| Name | Ball # | Type | Name | Ball # | Type | Name | Ball # | Type |
|------|--------|------|------|--------|------|------|--------|------|
| MAB0 | W12 | O | MD29 | A17 | I/O | MD60 | A18 | I/O |
| MAB1 | U12 | O | MD30 | D16 | I/O | MD61 | B17 | I/O |
| MD00 | W20 | I/O | MD31 | B14 | I/O | MD62 | A16 | I/O |
| MD01 | T17 | I/O | MD32 | V19 | I/O | MD63 | C15 | I/O |
| MD02 | T19 | I/O | MD33 | U19 | I/O | MPD0 | H19 | I/O |
| MD03 | R20 | I/O | MD34 | T20 | I/O | MPD1 | G19 | I/O |
| MD04 | P19 | I/O | MD35 | R17 | I/O | MPD2 | J18 | I/O |
| MD05 | N18 | I/O | MD36 | P18 | I/O | MPD3 | F20 | I/O |
| MD06 | L20 | I/O | MD37 | M20 | I/O | MPD4 | H18 | I/O |
| MD07 | K19 | I/O | MD38 | L19 | I/O | MPD5 | G20 | I/O |
| MD08 | F18 | I/O | MD39 | K18 | I/O | MPD6 | H20 | I/O |
| MD09 | D20 | I/O | MD40 | F19 | I/O | MPD7 | G18 | I/O |
| MD10 | C19 | I/O | MD41 | E18 | I/O | MWE# | T11 | O |
| MD11 | C18 | I/O | MD42 | C20 | I/O | NA# | N5 | O |
| MD12 | B18 | I/O | MD43 | A20 | I/O | PAR | E12 | I/O |
| MD13 | B16 | I/O | MD44 | D17 | I/O | PCLKIN | E10 | I |
| MD14 | B15 | I/O | MD45 | C16 | I/O | PHLD# | D4 | I |
| MD15 | C13 | I/O | MD46 | D15 | I/O | PHLDA# | D5 | O |
| MD16 | U18 | I/O | MD47 | C14 | I/O | RAS0# | G17 | O |
| MD17 | U20 | I/O | MD48 | V20 | I/O | RAS1# | F17 | O |
| MD18 | R18 | I/O | MD49 | T18 | I/O | RAS2# | F16 | O |
| MD19 | P17 | I/O | MD50 | R19 | I/O | RAS3# | E16 | O |
| MD20 | N20 | I/O | MD51 | P20 | I/O | RAS4# | M17 | O |
| MD21 | M19 | I/O | MD52 | N19 | I/O | RAS5# | N17 | O |
| MD22 | L18 | I/O | MD53 | M18 | I/O | RAS6# | M16 | O |
| MD23 | J 20 | I/O | MD54 | K20 | I/O | RAS7# | N16 | O |
| MD24 | E19 | I/O | MD55 | J19 | I/O | REQ0# | D7 | O |
| MD25 | E17 | I/O | MD56 | E20 | I/O | REQ1# | D9 | O |
| MD26 | D18 | I/O | MD57 | D19 | I/O | REQ2# | D11 | O |
| MD27 | A19 | I/O | MD58 | B20 | I/O | REQ3# | D13 | O |
| MD28 | C17 | I/O | MD59 | B19 | I/O | RST# | T15 | I |

| Name | Ball # | Type | | Name | Ball # | Type | | Name | Ball # | Type |
|---|---|---|---|---|---|---|---|---|---|---|
| SERR# | E13 | O | | TWE# | V16 | O | | VSS | M9 | — |
| SMIACT# | T10 | I | | VCC3 | R16 | — | | VSS | J 10 | — |
| STOP# | E11 | I/O | | VCC3 | F5 | — | | VSS | K10 | — |
| TEST# | W17 | I | | VCC3 | F6 | — | | VSS | L10 | — |
| TIO0 | P16 | I/O | | VCC3 | G6 | — | | VSS | M10 | — |
| TIO1 | U16 | I/O | | VCC3 | R6 | — | | VSS | J 11 | — |
| TIO2 | U17 | I/O | | VCC3 | R7 | — | | VSS | K11 | — |
| TIO3 | V17 | I/O | | VCC3 | F14 | — | | VSS | L11 | — |
| TIO4 | T14 | I/O | | VCC3 | F15 | — | | VSS | M11 | — |
| TIO5 | V18 | I/O | | VCC3 | P15 | — | | VSS | J 12 | — |
| TIO6 | W18 | I/O | | VCC3 | R15 | — | | VSS | K12 | — |
| TIO7 | Y18 | I/O | | VDD5 | E14 | — | | VSS | L12 | — |
| TIO8 | W19 | I/O | | VSS | T6 | — | | VSS | M12 | — |
| TIO9 | Y | I/O | | VSS | J 9 | — | | VSS | E15 | — |
| TIO10 | Y20 | I/O | | VSS | K9 | — | | VSS | T16 | — |
| TRDY# | E8 | I/O | | VSS | L9 | — | | W/R# | T9 | I |

intel®

## 5.2. TXC Package Information

This specification outlines the mechanical dimensions for the TXC. The package is a 324 pin ball grid array (BGA).
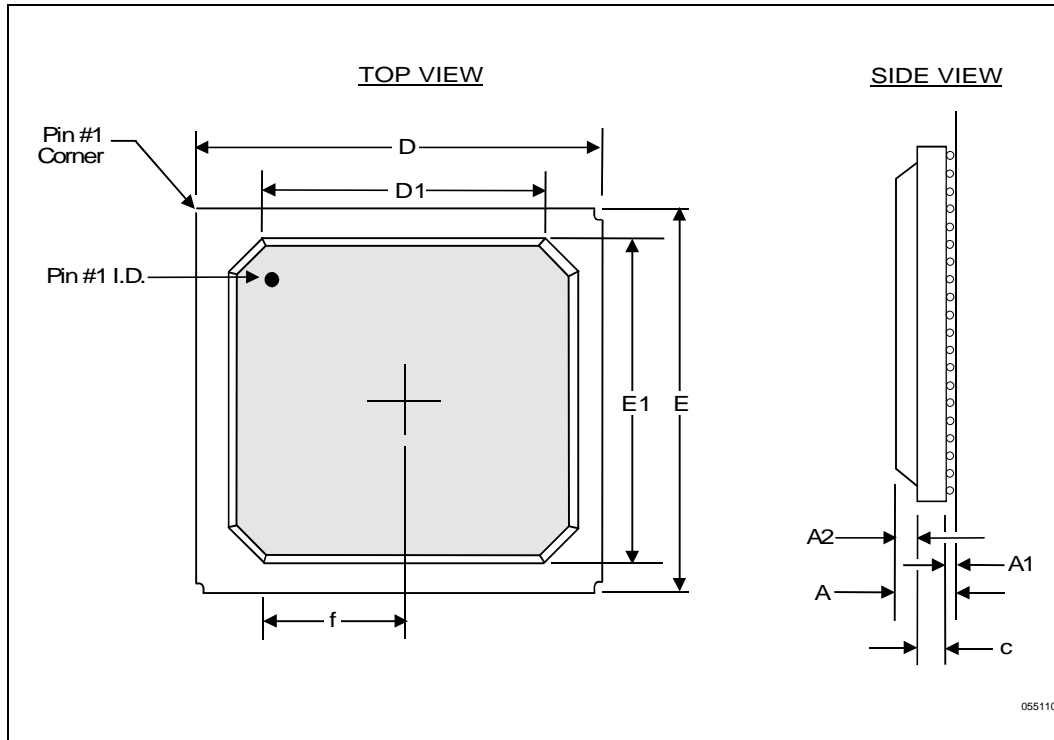


**Figure 9.  TXC 324 pin ball grid array (BGA).**
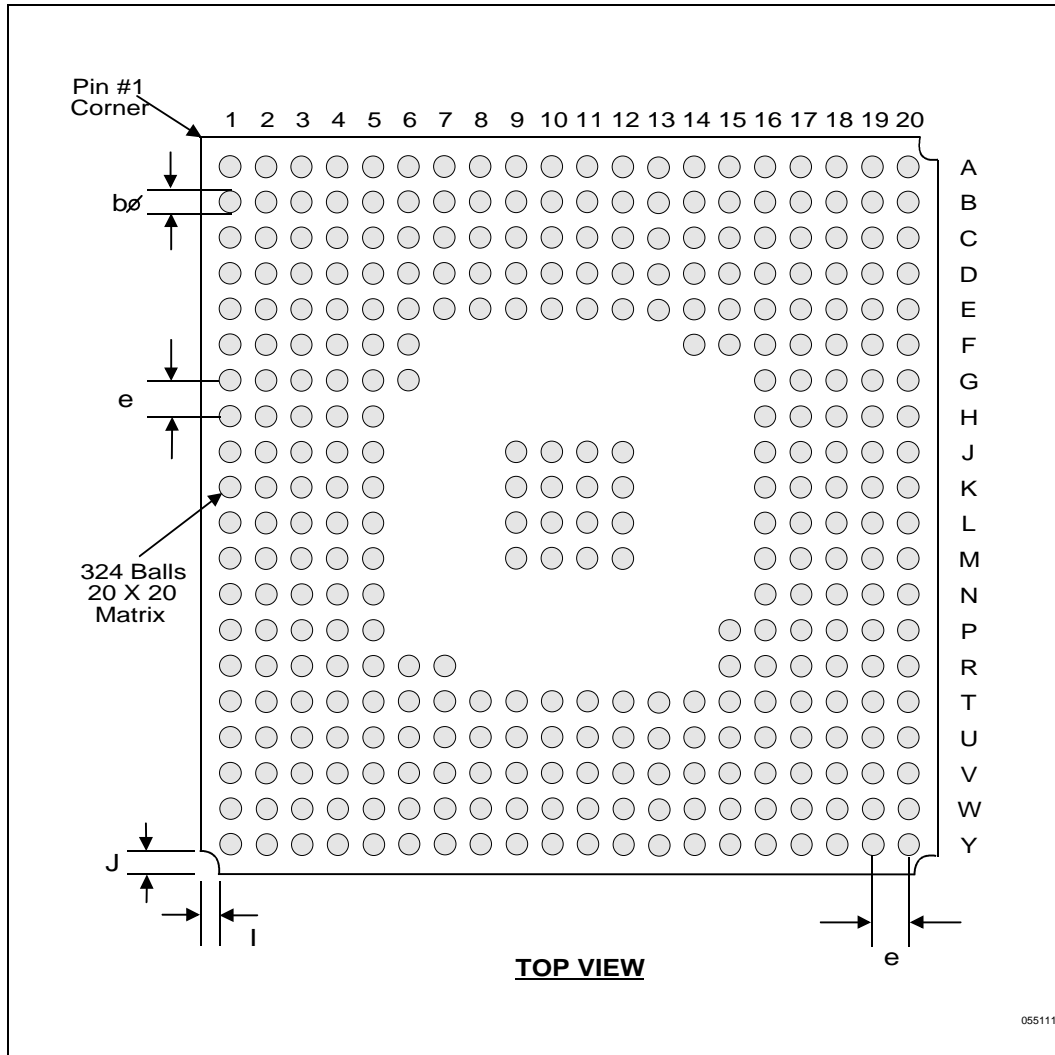
**Figure 10.  TXC 324 pin ball grid array (BGA) Ball Pattern**

**Table 17.  TXC 324 pin ball grid array (BGA).**

| SYMBOL | e=1.27 (solder ball pitch) | | | NOTE |
|---|---|---|---|---|
| | **MIN.** | **NOM.** | **MAX.** | 1 |
| A | 1.90(2.06) | 2.09(2.29) | 2.30(2.52) | |
| A1 | 0.50 | 0.60 | 0.70 | |
| A2 | 1.12 | 1.17 | 1.22 | |
| D | 26.80 | 27.00 | 27.20 | |
| D1 | | 24.00 | 24.70 | |
| E | 26.80 | 27.00 | 27.20 | |
| E1 | | 24.00 | 24.70 | |
| I | 1.44 REF. | | | |
| J | 1.44 REF. | | | |
| M | 20 (DEPOPULATED) | | | 2 |
| N | 324 | | | 3 |
| b | 0.60 | 0.76 | 0.90 | |
| c | 0.28(0.44) | 0.32(0.52) | 0.38(0.60) | |
| f | 8.05 REF. | | | |
| REMARK | 2 LAYER (4 LAYER) | | | |

**NOTES:**

1.  All dimensions are in millimeters.
2.  'M' represents the maximum solder ball matrix size.
3.  'N' Represents the maximum allowable number of solder balls.

intel®

## 6.0.    TESTABILITY

### 6.1.    NAND Tree Mode

A NAND tree mode is provided  for Automated Test Equipment (ATE) board level testing. The NAND tree allows the tester to set the connectivity of each of the TXC's signal pins. The NAND tree mode is activated by driving the test-pin TEST# when REQ[3:0]# pins are 0010. If TEST is negated at any time, the test-mode is de-activated and the TXC goes back to normal operation. There is no guarantee that upon re-entering normal operation the chip will function properly if the test-mode was entered while the TXC was not in a completely idle state.

In NAND tree mode, all outputs are Tri-stated except GNT[3:0]# and PHLDA#. These pins are the NAND-chain and test mode outputs.  The chain is separated into 4 77-pin chains.

#### 6.1.1.    OVERVIEW

The TXC has 2 full-chip test-modes to improve board manufacture. If the TEST# is asserted (driven low), the value on REQ#[3:0] indicates the test-mode to enable.  The test-mode enabled at the rising edge of TEST# will remain enabled while TEST# is high (Figure 11).  A new test-mode can be enabled by again driving TEST# low and selecting a new combination on PHLD# and REQ#[3:0]. Any active test-mode is disabled by asserting RST# with TEST# is de-asserted  or selecting the DISABLE mode with TEST# low (see table below).  Asserting RST# while TEST# is asserted has no effect on the test-mode.  In addition, all outputs are floated while RST# and TEST# are both asserted.
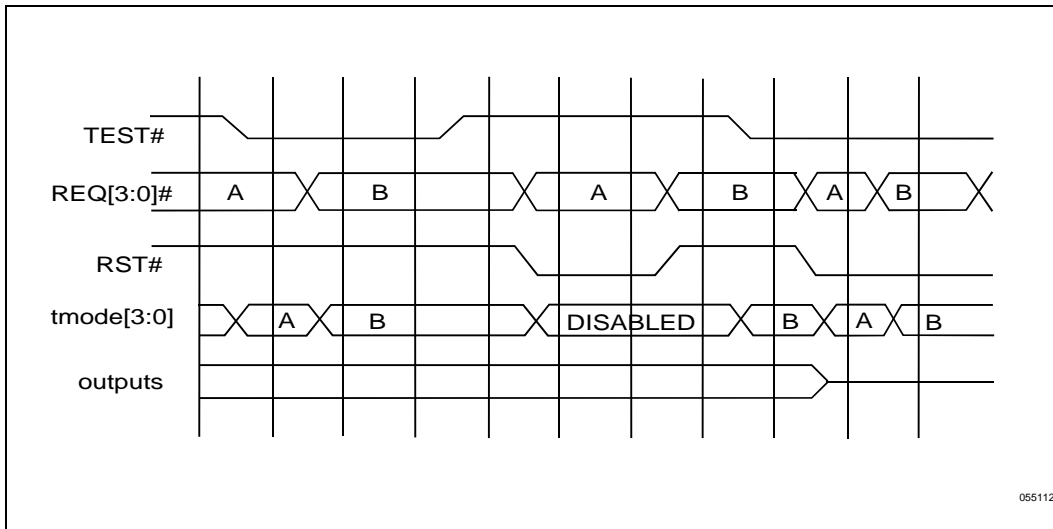


**Figure 11.  Full-Chip Test Mode Timing**

Table 18 shows each test-mode and the value of REQ[3:0]# required to enable it. All other values of REQ[3:0]# while TEST# is asserted are reserved and should not be asserted by the customer.

**Table 18.  Test Mode REQx Values**

| REQ[3:0]# | Test Mode | Description |
|---|---|---|
| 0010 | NAND Chain | Float All Outputs, Enable 4 NAND-Chains on GNT[3:0]# |
| 1110 | ID Code | Drives Device ID and Revision ID ON AD[15:0] |
| 1111 | Disable Test Modes | Disables any active test-mode.  Puts TXC back into normal operation. |

The following sub-sections describe both test-modes in more detail.

### 6.1.2.    NAND CHAIN MODE

All outputs are tri-stated, except GNT[3:0]#. All pins on the TXC are either pure inputs are bi-directional. Because the bi-directional pins are tri-stated during nand-chain test there is no need to back-drive outputs. However, all internal pull-ups or pull-downs are still active.  The GNT[3:0]# pins drive the output of the NAND Chains. The only pins not included on the NAND-Chain are RST#, TEST# and GNT[3:0]#. The RST# pin is NOT required for enabling NAND-Chain mode.  This is important to board-level test as assertion of reset on a system board can force other components into an undesirable state. All pins must be driven using a maximum of 3.3V.

The following I/O pins are not included on any of these NAND-chains:

**Excluded Pins:**
GNT[3:0]#
TEST#
RST#

The remaining 288 pins are split into 4 chains of 72 pins each. These chains are not chained together in any way.  The following tables show the pin ordering for each NAND-chain and the GNT# pin it is driven out on.

**Table 19. Chain 0 (GNT0#)**

| Pin Name | Chain Element | Pin Name | Chain Element | Pin Name | Chain Element |
|---|---|---|---|---|---|
| A25 | CH0_00 | HD8 | CH0_25 | HLOCKB | CH0_50 |
| A7 | CH0_01 | HD19 | CH0_26 | HD53 | CH0_51 |
| A10 | CH0_02 | HD7 | CH0_27 | HD60 | CH0_52 |
| A21 | CH0_03 | HD31 | CH0_28 | HD62 | CH0_53 |
| A23 | CH0_04 | HD26 | CH0_29 | HD61 | CH0_54 |
| SMIACTB | CH0_05 | HD27 | CH0_30 | AD30 | CH0_55 |
| A31 | CH0_06 | HD33 | CH0_31 | AD25 | CH0_56 |
| A18 | CH0_07 | AHOLD | CH0_32 | AD22 | CH0_57 |
| HD3 | CH0_08 | BE5B | CH0_33 | AD24 | CH0_58 |
| A12 | CH0_09 | BE6B | CH0_34 | AD27 | CH0_59 |
| A14 | CH0_10 | HD34 | CH0_35 | AD19 | CH0_60 |
| HD11 | CH0_11 | BE2B | CH0_36 | C_BE3B | CH0_61 |
| A19 | CH0_12 | BE0B | CH0_37 | REQ0B | CH0_62 |
| A16 | CH0_13 | HD37 | CH0_38 | AD20 | CH0_63 |
| HITMB | CH0_14 | BE1B | CH0_39 | AD21 | CH0_64 |
| HD13 | CH0_15 | HD36 | CH0_40 | TRDYB | CH0_65 |
| D_CB | CH0_16 | HD41 | CH0_41 | AD16 | CH0_66 |
| HD0 | CH0_17 | HD52 | CH0_42 | C_BE2B | CH0_67 |
| HD10 | CH0_18 | CACHEB | CH0_43 | C_BE1B | CH0_68 |
| HD16 | CH0_19 | HD47 | CH0_44 | AD14 | CH0_69 |
| HD6 | CH0_20 | HD44 | CH0_45 | AD12 | CH0_70 |
| HD14 | CH0_21 | HD49 | CH0_46 | AD13 | CH0_71 |
| BOFFB | CH0_22 | HD54 | CH0_47 | | |
| HD18 | CH0_23 | HD55 | CH0_48 | | |
| HD24 | CH0_24 | HD57 | CH0_49 | | |

**Table 20.  Chain 1 (GNT1#)**

| Pin Name | Chain Element | Pin Name | Chain Element | Pin Name | Chain Element |
|---|---|---|---|---|---|
| A24 | CH1_00 | HD21 | CH1_25 | M_IOB | CH1_50 |
| A27 | CH1_01 | NAB | CH1_26 | HD59 | CH1_51 |
| A26 | CH1_02 | HD25 | CH1_27 | HD63 | CH1_52 |
| A13 | CH1_03 | BRDYB | CH1_28 | PHLDB | CH1_53 |
| A8 | CH1_04 | HD23 | CH1_29 | AD31 | CH1_54 |
| W_RB | CH1_05 | HD35 | CH1_30 | PLOCKB | CH1_55 |
| A11 | CH1_06 | HD29 | CH1_31 | AD28 | CH1_56 |
| A17 | CH1_07 | HD30 | CH1_32 | FRAMEB | CH1_57 |
| A15 | CH1_08 | BE7B | CH1_33 | AD29 | CH1_58 |
| A5 | CH1_09 | HD32 | CH1_34 | PHLDAB | CH1_59 |
| A9 | CH1_10 | BE3B | CH1_35 | IRDYB | CH1_60 |
| HD1 | CH1_11 | BE4B | CH1_36 | AD26 | CH1_61 |
| HD9 | CH1_12 | HD39 | CH1_37 | AD23 | CH1_62 |
| A20 | CH1_13 | HD45 | CH1_38 | AD17 | CH1_63 |
| HD2 | CH1_14 | HD42 | CH1_39 | DEVSELB | CH1_64 |
| HD5 | CH1_15 | HD38 | CH1_40 | AD18 | CH1_65 |
| HD4 | CH1_16 | HD40 | CH1_41 | REQ1B | CH1_66 |
| ADSB | CH1_17 | HD48 | CH1_42 | AD15 | CH1_67 |
| HD15 | CH1_18 | KENB | CH1_43 | PCLKIN | CH1_68 |
| EADSB | CH1_19 | HD46 | CH1_44 | AD11 | CH1_69 |
| HD20 | CH1_20 | HD43 | CH1_45 | AD10 | CH1_70 |
| HD17 | CH1_21 | HD51 | CH1_46 | AD9 | CH1_71 |
| HD22 | CH1_22 | HD58 | CH1_47 | | |
| HD12 | CH1_23 | HD56 | CH1_48 | | |
| HD28 | CH1_24 | HD50 | CH1_49 | | |

**Table 21.  Chain 2 (GNT2#)**

| Pin Name | Chain Element | Pin Name | Chain Element | Pin Name | Chain Element |
|---|---|---|---|---|---|
| A28 | CH2_00 | MD2 | CH2_25 | MD25 | CH2_50 |
| A29 | CH2_01 | MD50 | CH2_26 | MD42 | CH2_51 |
| A3 | CH2_02 | MD34 | CH2_27 | RAS2B | CH2_52 |
| A30 | CH2_03 | RAS6B | CH2_28 | MD59 | CH2_53 |
| A4 | CH2_04 | MD5 | CH2_29 | MD11 | CH2_54 |
| MWEB | CH2_05 | CAS7B | CH2_30 | MD43 | CH2_55 |
| MA10 | CH2_06 | RAS4B | CH2_31 | MD27 | CH2_56 |
| MAA1 | CH2_07 | MD53 | CH2_32 | MD30 | CH2_57 |
| MA7 | CH2_08 | MD22 | CH2_33 | MD28 | CH2_58 |
| MA5 | CH2_09 | MD6 | CH2_34 | MD12 | CH2_59 |
| MA11 | CH2_10 | MD39 | CH2_35 | MD60 | CH2_60 |
| MA6 | CH2_11 | MD7 | CH2_36 | AD2 | CH2_61 |
| MA9 | CH2_12 | MD54 | CH2_37 | STOPB | CH2_62 |
| BWEB | CH2_13 | HCLKIN | CH2_38 | AD4 | CH2_63 |
| COEB | CH2_14 | MD55 | CH2_39 | AD0 | CH2_64 |
| CCSB | CH2_15 | MPD2 | CH2_40 | AD3 | CH2_65 |
| MD0 | CH2_16 | CAS5B | CH2_41 | AD5 | CH2_66 |
| MD1 | CH2_17 | MPD1 | CH2_42 | AD7 | CH2_67 |
| MD16 | CH2_18 | CAS2B | CH2_43 | AD6 | CH2_68 |
| MD32 | CH2_19 | MPD7 | CH2_44 | C_BE0B | CH2_69 |
| MD48 | CH2_20 | CAS4B | CH2_45 | REQ2B | CH2_70 |
| MD49 | CH2_21 | MPD3 | CH2_46 | AD8 | CH2_71 |
| MD18 | CH2_22 | MD40 | CH2_47 |  |  |
| MD19 | CH2_23 | RAS0B | CH2_48 |  |  |
| MD17 | CH2_24 | MD26 | CH2_49 |  |  |

**Table 22. Chain 3 (GNT3#)**

| Pin Name | Chain Element | Pin Name | Chain Element | Pin Name | Chain Element |
|---|---|---|---|---|---|
| A22 | CH3_00 | MD35 | CH3_25 | MD41 | CH3_50 |
| MAB0 | CH3_01 | MD33 | CH3_26 | MD57 | CH3_51 |
| MAA0 | CH3_02 | MD36 | CH3_27 | CAS6B | CH3_52 |
| A6 | CH3_03 | RAS5B | CH3_28 | MD10 | CH3_53 |
| MA3 | CH3_04 | MD4 | CH3_29 | RAS3B | CH3_54 |
| MA2 | CH3_05 | MD3 | CH3_30 | MD58 | CH3_55 |
| MA4 | CH3_06 | MD52 | CH3_31 | MD44 | CH3_56 |
| MAB1 | CH3_07 | MD51 | CH3_32 | SERRB | CH3_57 |
| CADSB | CH3_08 | MD20 | CH3_33 | MD46 | CH3_58 |
| MA8 | CH3_09 | MD21 | CH3_34 | MD45 | CH3_59 |
| GWEB | CH3_10 | MD37 | CH3_35 | MD61 | CH3_60 |
| CADVB | CH3_11 | MD38 | CH3_36 | MD31 | CH3_61 |
| TWEB | CH3_12 | CAS3B | CH3_37 | MD29 | CH3_62 |
| TIO4 | CH3_13 | CAS1B | CH3_38 | MD13 | CH3_63 |
| TIO3 | CH3_14 | MD23 | CH3_39 | MD15 | CH3_64 |
| TIO1 | CH3_15 | MPD0 | CH3_40 | REQ3B | CH3_65 |
| TIO7 | CH3_16 | MPD6 | CH3_41 | MD63 | CH3_66 |
| TIO6 | CH3_17 | MPD4 | CH3_42 | MD62 | CH3_67 |
| TIO9 | CH3_18 | CAS0B | CH3_43 | MD47 | CH3_68 |
| TIO2 | CH3_19 | MPD5 | CH3_44 | PAR | CH3_69 |
| TIO8 | CH3_20 | MD56 | CH3_45 | MD14 | CH3_70 |
| TIO5 | CH3_21 | MD8 | CH3_46 | AD1 | CH3_71 |
| TIO10 | CH3_22 | MD9 | CH3_47 | | |
| TIO0 | CH3_23 | RAS1B | CH3_48 | | |
| RAS7B | CH3_24 | MD24 | CH3_49 | | |

## 6.2.    ID Code Test Mode

To enable this mode TEST# and RST# must be asserted and REQ[3:0]# must be set to 1110.  RST# is then negated after 2 HCLKs and TEST# must then be negated after 2 more HCLKs and REQ[3:0]# changed to the value 1111.  The value of PHLD# and RST# pin must also be 1. Once REQ[3:0]#, PHLD# and RST# are all driven high, the Device ID (DID) will be driven on AD[15:0], and the Revision ID (RID) will be driven on AD[23:16].  The value on AD[31:24] is indeterminate.  The following timing diagram shows how the test-mode is enabled:
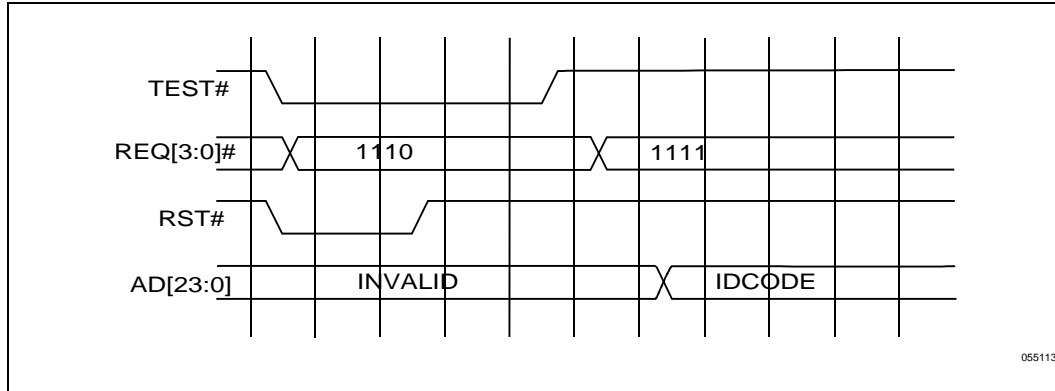


**Figure 12.  Test Mode Enable**