MS7AA Memory Technical Manual

Order Number EK–MS7AA–TM.001

The MS7AA memory module is designed for computer systems built around the LSB bus. The module can be configured in incremental storage capacities ranging from 64 Mbytes to 2 Gbytes.

digital equipment corporation
maynard, massachusetts

First Printing, November 1992

FCC NOTICE: The equipment described in this manual generates, uses, and may emit radio frequency energy. The equipment has been type tested and found to comply with the limits for a Class A computing device pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such radio frequency interference when operated in a commercial environment. Operation of this equipment in a residential area may cause interference, in which case the user at his own expense may be required to take measures to correct the interference.

Contents

Chapter 5 LSB Memory Signals

Chapter 6 Memory Transactions

Chapter 7 Registers

Chapter 8 Error Conditions

Chapter 9 Self-Test

Figures

Tables

Preface

Intended Audience

This manual describes the bit- level hardware functions of the MS7AA memory module used in computer systems designed around the LSB bus. It is written for developers of system and application software. It assumes machine level programming knowledge and familiarity with the operating system.

Document Structure

The material is presented in nine chapters.

- Chapter 1, **Overview,** presents an overall introduction to the MS7AA memory module.
- Chapter 2, **Memory Interface Controller,** describes the hardware interface to the LSB bus and discusses the functions of the logic elements implemented on the MIC chips.
- Chapter 3, **Memory Data Controller,** describes the interface between the MIC chips and the dynamic random access memory array of the memory module.
- Chapter 4, **Memory Organization,** describes how memory is organized on a single module or multiple modules and explains how memory performance is enhanced by dividing the DRAMs into two banks and using interleaving techniques.
- Chapter 5, **LSB Memory Signals,** describes the signals used by the memory module to communicate with the LSB bus.
- Chapter 6, **Memory Transactions,** describes how read and write operations are executed on the memory modules.
- Chapter 7, **Registers,** lists the LSB- required and memory- specific registers and provides bit- level functional descriptions of each register.
- Chapter 8, **Error Conditions,** describes the two categories of errors detected by the memory module and explains how errors are reported in associated error syndrome registers.
- Chapter 9, **Self-Test,** tells when a memory module performs a self- test and gives a brief overview of functions that the memory self- test provides.

Conventions Used in This Document

Results and Operations

Results of operations termed UNPREDICTABLE must not be used by software.

Operations termed UNDEFINED do not cause the processor to hang, that is, reach a state from which there is no transition to a normal state of instruction execution. Nonprivileged software cannot invoke UNDEFINED operations.

Register and Bit Designations

Certain conventions are followed in register descriptions and in references to bits and bit fields:

- Registers are referred to by their mnemonics, such as **MCR register.** The full name of a register (for example, **Memory Configuration Register**) is spelled out only at the top of the register description page, or when the register is first introduced.
- Bits and fields are enclosed in angle brackets. For example, **bit <31>** and **bits <31:16>**. For clarity of reference, bits are usually specified by their numbers or names enclosed in angle brackets adjacent to the register mnemonic, such as **AMR<31:17>** or **AMR<MADR>,** which are equivalent designations.
- When the value of a bit position is given explicitly in a register diagram, the information conveyed is as follows:

Bit Value Designation	Meaning
0	Reads as zero; ignored on writes.
1	Reads as one; ignored on writes.
X	Does not exist in hardware. The value of the bit is unpredictable on reads and ignored on writes.

- Fields noted as must be zero (MBZ) must never be filled by software with a nonzero value.
- The entry in the **type** column of a register description table may include the initialization value of the bits. For example, entry "R/W, 0" indicates a read/write bit that is initialized to 0.

Terminology

Table 1 gives the definition of terms and some mnemonics used in this document.

Table 1 Definitions of Terms Used in This Document

Term	Definition
Bank	Smallest group of DRAMs that can be interleaved. A bank consists of one or more *strings*.
Block	64 bytes of data within naturally aligned boundaries.
DDB	DRAM data bus. The 576- bit bidirectional data bus that interfaces between the DRAM chips and the MDC gate arrays.
LSB	The system bus.
MDB	Memory data bus. The 144- bit bidirectional data bus is the interface between the MIC gate arrays and the MDC gate arrays.
MDC	Memory data controller. Chips that buffer data between the MIC and the DRAM arrays.
MIC	Memory interface controller. The interface of the memory subsystem to the LSB.
String	The smallest group of DRAMs (144 1/4M x 4) needed to store and retrieve 64 bytes of data per LSB transaction. In some array implementations, the number of banks and strings can be equal, while in others there may be more strings than banks.
32- bit ECC	Synonymous with longword ECC.
64- bit ECC	Synonymous with quadword ECC.

Documentation Titles

Table 2 lists the books in the DEC 7000/10000 and VAX 7000/10000 documentation sets. Table 3 lists other documents that you may find useful.

Table 2 DEC 7000/10000 and VAX 7000/10000 Documentation

Title	7000 Systems Order Number	10000 Systems Order Number
Installation Kit	EK–7000B–DK	EK–1000B–DK
Site Preparation Guide	EK–7000B–SP	EK–1000B–SP
Installation Guide	EK–700EB–IN	EK–100EB–IN
Hardware User Information Kit	EK–7001B–DK	EK–1001B–DK
Operations Manual	EK–7000B–OP	EK–1000B–OP
Basic Troubleshooting	EK–7000B–TS	EK–1000B–TS
Service Information Kit—VAX 7000	EK–7002A–DK	EK–1002A–DK
Platform Service Manual	EK–7000A–SV	EK–1000A–SV
System Service Manual	EK–7002A–SV	EK–1002A–SV
Pocket Service Guide	EK–7000A–PG	EK–1000A–PG
Advanced Troubleshooting	EK–7001A–TS	EK–1001A–TS
Service Information Kit—DEC 7000	EK–7002B–DK	EK–1002A–DK
Platform Service Manual	EK–7000A–SV	EK–1000A–SV
System Service Manual	EK–7002B–SV	EK–1002A–SV
Pocket Service Guide	EK–7700A–PG	EK–1100A–PG
Advanced Troubleshooting	EK–7701A–TS	EK–1101A–TS
Reference Manuals		
Console Reference Manual	EK–70C0B–TM	
KA7AA CPU Technical Manual	EK–KA7AA–TM	
KN7AA CPU Technical Manual	EK–KN7AA–TM	
MS7AA Memory Technical Manual	EK–MS7AA–TM	
I/O System Technical Manual	EK–70I0A–TM	
Platform Technical Manual	EK–7000A–TM	

Table 2 DEC 7000/10000 and VAX 7000/10000 Documentation (Continued)

Title	7000 Systems Order Number	10000 Systems Order Number
Upgrade Manuals		
KA7AA CPU Installation Guide	EK–KA7AA–IN	
KN7AA CPU Installation Guide	EK–KN7AA–IN	
MS7AA Memory Installation Guide	EK–MS7AA–IN	
KZMSA Adapter Installation Guide	EK–KXMSX–IN	
DWLMA XMI PIU Installation Guide	EK–DWLMA–IN	
DWMBB VAXBI PIU Installation Guide	EK–DWMBB–IN	
H7237 Battery PIU Installation Guide	EK–H7237–IN	
H7263 Power Regulator Installation Guide	EK–H7263–IN	
Futurebus+ PIU Installation Guide	EK–DWLAA–IN	
BA654 DSSI Disk PIU Installation Guide	EK–BA654–IN	
BA655 SCSI Disk and Tape PIU Installation Guide	EK–BA655–IN	
Removable Media Installation Guide	EK–TFRRD–IN	

Table 3 Related Documents

Title	Order Number
General Site Preparation	
Site Environmental Preparation Guide	EK–CSEPG–MA
System I/O Options	
BA350 DECstor/me Modular Storage Shelf Subsystem Configuration Guide	EK–BA350–CG
BA350 DECstor/me Modular Storage Shelf Subsystem User's Guide	EK–BA350–UG
BA350-LA DECstor/me Modular Storage Shelf User's Guide	EK–350LA–UG
CIXCD Interface User Guide	EK–CIXCD–UG
DEC FDDIcontroller 400 Installation/Problem Solving	EK–DEMFA–IP
DEC LANcontroller 400 Installation Guide	EK–DEMNA–IN
DEC LANcontroller 400 Technical Manual	EK–DEMNA–TM
DSSI VAXcluster Installation and Troubleshooting Manual	EK–410AA–MG
InfoServer 150 Installation and Owner's Guide	EK–INFSV–OM
KDM70 Controller User Guide	EK–KDM70–UG
KFMSA Module Installation and User Manual	EK–KFMSA–IM
KFMSA Module Service Guide	EK–KFMSA–SV
RRD42 Disc Drive Owner's Manual	EK–RRD42–OM
RF Series Integrated Storage Element User Guide	EK–RF72D–UG
TF85 Cartridge Tape Subsystem Owner's Manual	EK–OTF85–OM
TLZ06 Cassette Tape Drive Owner's Manual	EK–TLZ06–OM
Operating System Manuals	
Alpha Architecture Reference Manual	EY–L520E–DP
DEC OSF/1 Guide to System Administration	AA–PJU7A–TE
DECnet for OpenVMS Network Management Utilities	AA–PQYAA–TK
Guide to Installing DEC OSF/1	AA–PS2DA–TE
OpenVMS Alpha Version 1.0 Upgrade and Installation Manual	AA–PQYSA–TE
VMS Upgrade and Installation Supplement: VAX 7000–600 and VAX 10000–600 Series	AA–PRAHA–TE
VMS Network Control Program Manual	AA–LA50A–TE

Table 3 Related Documents (Continued)

Title	Order Number
VMSclusters and Networking	
HSC Installation Manual	EK–HSCMN–IN
SC008 Star Coupler User's Guide	EK–SC008–UG
VAX Volume Shadowing Manual	AA–PBTVA–TE
Peripherals	
Installing and Using the VT420 Video Terminal	EK–VT420–UG
LA75 Companion Printer Installation and User Guide	EK–LA75X–UG

Chapter 1

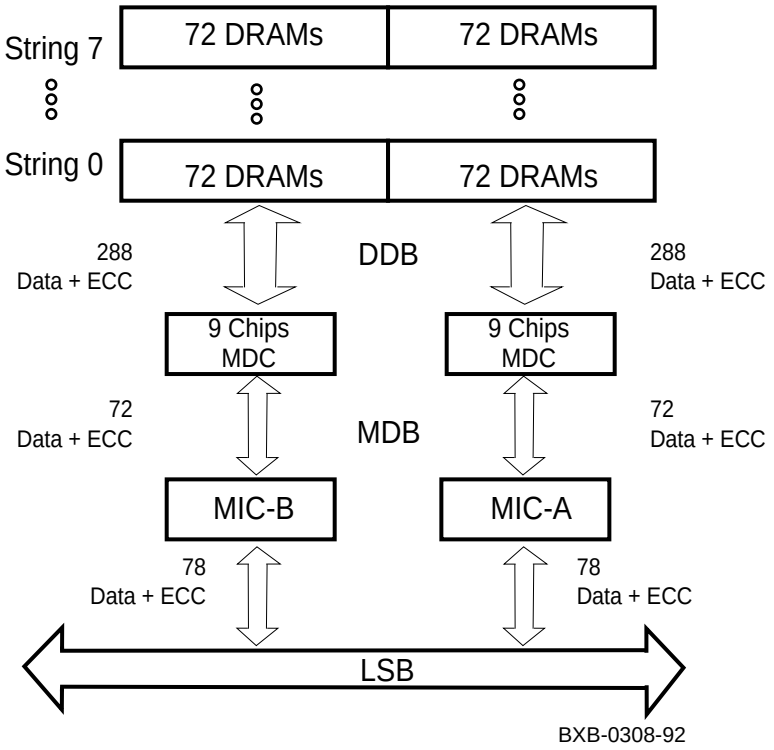
Overview

The MS7AA memory module connects directly to the LSB bus and provides up to 2 Gbytes of dynamic random access memory (DRAM) to the CPU. The LSB memory subsystem features the following:

- 64- Mbyte to 2- Gbyte memory capacity per module
- Incremental configuration to a maximum of seven modules in a single-processor system
- 1- , 2- , 4- , and 8- way interleaving
- 64- byte block transfers, 16 bytes at a time, over the system bus
- Memory modules with DRAM arrays of 1M x 4 or 4M x 4 components
- Bit scattering to provide correction of data if a single DRAM fails
- Read and write data wrapping on 32- byte naturally aligned boundaries
- Quadword ECC protection that allows detection and correction of single- bit failures and detection of 2- bit failures
- Conversion logic that translates quadword ECC (DRAM ECC) to longword ECC (LSB ECC) during memory reads and longword ECC to quadword ECC during memory writes

Figure 1- 1 shows the architecture of a memory module.

Figure 1-1 Memory Module Block Diagram



1.1 Major Sections

An MS7AA module consists of three major sections:

- Memory interface controller (MIC)
- Memory data controller (MDC)
- DRAM arrays

The major sections communicate with each other through internal buses.

1.1.1 Memory Interface Controller

The memory interface controller (MIC) is comprised of two gate arrays, MIC- A and MIC- B. It provides the interface to the LSB, controls DRAM timing and refresh, runs memory self- test, and contains all LSB- required and memory- specific registers.

The MIC- A interfaces to LSB D<63:0> and the LSB control signals. It decodes the LSB command and determines if the memory module is selected for this transaction. The MIC- A provides the address and control signals for the two banks and contains most of the CSRs.

The MIC- B interfaces to LSB D<127:64>. It is strictly a data path, with some error checking logic and MIC- B- specific CSRs. The operation of the MIC- B is controlled by the MIC- A.

1.1.2 Memory Data Controller

The memory data controller (MDC) consists of DRAM buffers that reside between the MIC and the DRAM arrays. During a memory write, the MDC buffers accumulate the four LSB data cycles into a 64- byte packet that is written to the DRAMs in a single operation. During a memory read, the DRAM buffers store 64 bytes of data, then parcel it to the MIC, which in turn parcels it out to the LSB in four consecutive 16- byte cycles under the control of the MIC- A.

1.1.3 DRAM Arrays

The DRAM arrays consist of DRAMs, control signal, and address buffer components. The MS7AA memory modules can use DRAM sizes of 1M x 4 bits or 4M x 4 bits. The DRAM arrays are organized into 1 to 8 strings. Each string requires 144 DRAMs (using DRAMs with quadword ECC), regardless of the DRAM type. A single string supports 64 Mbytes of memory when configured with 4- Mbit DRAMs, and 256 Mbytes of memory when configured with 16- Mbit DRAMs. The DRAM array on each memory module is configured with two independently accessible banks.

Interleaving of DRAM banks increases memory bandwith. Each memory module supports 2- way interleaving when configured with a minimum of two strings. Interleaving occurs between the two independently accessible banks within a module. A memory configuration on the LSB consisting of four memory modules, with at least two strings each, supports a maximum of 8- way interleaving.

1.2 Internal Buses

Data within the memory modules is transferred through two internal buses: the memory data bus (MDB) and the DRAM data bus (DDB). The MDB bus transfers data between the MIC chips and the MDC chips. The DDB bus transfers data between the MDC chips and the DRAM arrays.

1.3 Transactions

Memory responds to but cannot initiate LSB transactions. It responds to accesses to the memory space and to its own LSB node space. Memory does not respond to any transaction in LSB broadcast space.

Memory modules run synchronously with the LSB. Memory transfers consist of four contiguous, 16- byte data cycles, for a total of 64 bytes per transaction. Up to three pipelined transactions can be in progress on the LSB bus at a given time. Read and write data wrapping is supported on 32-byte naturally aligned boundaries.

1.4 Refresh

Each memory module implements DRAM refresh. Module refresh is reset and restarted under two different conditions: power- up and system reset. All modules with the same DRAM refresh rate requirements refresh at the same time.

Chapter 2

Memory Interface Controller

The memory interface controller (MIC) controls data movement between the memory DRAM arrays and the LSB bus.

2.1 Components

The MIC consists of two 64- bit gate arrays, MIC- A and MIC- B.

The MIC- A interfaces to LSB D<63:0>, LSB control/status signals, and LSB ECC<13:0>. It controls the operation of the memory module as well as the operation of the MIC- B. The MIC- A generates the DRAM address and timing functions necessary for all DRAM operations. It also generates all signals used to control the memory data controller (MDC) chips.

The MIC- B interfaces to LSB D<127:64> and LSB ECC<27:14>. It functions strictly as a data path consisting of read and write data buffers, ECC logic, and CSRs.

The MIC- A implements most of the LSB- required and memory- specific registers. When a MIC- B control/status register is accessed by the CPU, data is transferred over an 8- bit bidirectional data path connecting the MIC- A and the MIC- B.

2.2 Logic Elements

The MIC implements the following logic elements:

- Command decode logic
- Address decode logic
- DRAM address logic
- DRAM control logic
- MIC data path logic
- Data wrapping
- Refresh logic
- Self- test logic
- LSB state machine control logic

The MIC also contains logic to support the memory data controller (MDC) interface.

2.2.1 Command Decode Logic

The LSB command decode logic receives and latches D<37:0> into its address latch during LSB command/address cycles. D<37:35> contain the LSB command field. Table 2- 1 lists the LSB commands and the associated operations within the memory module.

Table 2- 1 LSB Commands for Memory Operations

D<37:35>	Command	Memory Operation
000	Read	Read memory at address specified on LSB D<34:0>.
001	Write	Write memory at address specified on LSB D<34:0>.
010	Reserved	Ignore command.
011	Victim Write	Write memory at address specified on LSB D<34:0>.
100	Read CSR	Read memory CSR at address specified on LSB D<34:0>.
101	Write CSR	Write memory CSR at address specified on LSB D<34:0>.
110	Reserved	Ignore command.
111	Private	Ignore command.

2.2.2 Address Decode Logic

Memory responds to two types of transactions:

- Node space CSR transactions
- Memory space transactions

Memory modules do not respond to transactions in LSB broadcast space.

2.2.2.1 Node Space CSR Transactions

Node space contains LSB required registers and memory- specific registers accessed on aligned 64- byte boundaries. The MIC decodes the LSB node space address and processes the request to the specified register.

All registers are 32 bits wide. Since not all fields within registers contain a valit bit, CSR transactions involving bit fields with no associated valid bits return UNPREDICTABLE data on reads. Writes to nonexistent bits have no effect on module operations.

The MIC acknowledges memory node space requests whether the addressed CSR actually exists or not. Writes to nonexistent CSRs result in UNDEFINED operations. Reads to nonexistent CSRs return UNPREDICTABLE data.

2.2.2.2 Memory Space Transactions

The address decode logic compares the LSB address with bits from the Address Mapping Register (AMR) to determine if the memory module is selected for the current transaction. Various fields in the AMR register allow selection of memory space address decode parameters as described in the discussion of the AMR register in Chapter 7.

To support two banks per module, the address decode logic implements two independent address paths.

2.2.3 DRAM Address Logic

The DRAM address logic generates row and column address bits for various memory configurations from the LSB address.

2.2.4 DRAM Control Logic

The timing of DRAM components requires that control signals, along with the row and column addresses, conform to strict parameters. Each memory bank implements a DRAM timing generator to ensure that all DRAM control signals are properly sequenced during LSB memory access, module refresh, and module self- test.

In addition to DRAM control, the two timing generators pass control information between each other. Since there is one common data path between the MIC and the MDCs, communication between the two DRAM timing generators is needed to prevent data path conflicts.

2.2.5 MIC Data Path Logic

The MIC data path can be broken down into two distinct blocks. The data path to/from the DRAMs and the data path to/from the CSRs.

The data path to/from memory consists of a write path and a read path. Within each path there are data buffers and ECC logic. The data path to/from the CSRs implements parity generation logic needed to generate LSB parity on returning CSR read data.

The MIC data path logic implements a 4 x 16 byte write buffer and a 4 x 16 byte read buffer that are used as temporary storage elements on the write data path and the read data path, respectively. Storage of LSB write data is necessary when the MDCs are busy with a previous write transaction. Storage of LSB read data is necessary when LSB STALL is asserted by a node other than the memory node supplying the requested read data. The buffers are designed to minimize delays on the data paths.

The LSB operates on a 32- bit ECC algorithm, while memory stores and retrieves data with a 64- bit ECC algorithm. The write data path logic checks, corrects, and then converts LSB longword (32- bit) ECC into quadword (64- bit) ECC that is stored in each memory module. Similarly, the read data path ECC logic checks, corrects, and then converts quadword (64- bit) ECC used by memory into longword (32- bit) ECC used by the LSB.

2.2.6 Data Wrapping

The data wrapping logic controls the alignment of data in the MDC chips. LSB supports read and write data wrapping on 32- byte boundaries. During an LSB command/address cycle, LSB D<0> is deasserted on all naturally aligned 64- byte block transactions. It is asserted when the commander node issues the second 32 bytes of data prior to the first 32 bytes on write transactions, and expects the second 32 bytes of data prior to the first 32 bytes on read transactions.

Table 2- 2 defines the wrapping of read or write data depending upon the state of LSB D<0> latched during a command/address cycle.

Table 2- 2 LSB Data Wrapping

LSB D<0>	Data Cycle Ordering
0	Octaword- 1, Octaword- 2, Octaword- 3, Octaword- 4
1	Octaword- 3, Octaword- 4, Octaword- 1, Octaword- 2

2.2.7 Refresh Logic

The refresh logic refreshes the memory module approximately every 15.6 microseconds regardless of DRAM type (4 Mbit, 16 Mbit). The refresh period is generated by a 10- bit divide- by counter from LSB clocks. The actual count value is 780 ticks. A given module can be refreshed at the nominal rate, 2 or 4 times the nominal rate as determined by bits in the Memory Diagnostic Register. Upon power- up or reset, the default refresh rate is the nominal value. Table 2- 3 lists the refresh rate at various LSB cycle times.

Table 2- 3 Refresh Rates at Various LSB Cycle Times

LSB Cycle Time (ns)	Refresh Rate
16	12.48
17	13.26
18	14.04
19	14.82
20	15.60

Each memory module is refreshed independently of the other memory modules. Both banks of a memory array (all strings) are refreshed simultaneously with one exception: when an LSB read or write transaction is currently being serviced by one of the banks, refresh of the bank may be held off until the transaction has completed. Refresh is performed on the memory bank immediately following the completion of the transaction.

2.2.8 Self- Test Logic

Upon power- up, system reset, or when EXST_A (bit <4> of the Memory Diagnostic Register A) is set by software, each module executes a self- test designed to test and initialize the DRAMs with good ECC. While self- test is executing, any reference to memory space is NO ACKed. CSR space can be accessed during self- test execution. However, writes to the Memory Diagnostic Register A (MDRA) and Memory Diagnostic Register B (MDRB) can result in UNPREDICTABLE behavior. See Chapter 9 for self- test times.

2.2.9 LSB State Machine Control Logic

The LSB state machine control logic controls the data path interface to/from the LSB and all CSR operations (read/write). This logic also starts the DRAM timing generators.

Chapter 3

Memory Data Controller

The memory data controller (MDC) is the data interface between the DRAMs and the memory interface controller (MIC). It connects the 576- bit DRAM data path to the 144- bit MIC data path. The MDC operates under the control of the MIC.

3.1 MDC Data Paths

The MDC provides separate read and write data paths to the DRAM array on one side and the MIC on the other.

Each MDC block is composed of eighteen MDC chips. Each MDC chip interfaces between a 32- bit bidirectional DDB (DRAM data bus) port and an 8- bit bidirectional MDB (MIC data bus) port. Both ports are controlled by the MIC- A.

3.2 MDC Transactions

MDC supports four types of transactions:

- Read
- Wrapped read
- Write
- Wrapped write

3.2.1 Read and Wrapped Read Transactions

During a read transaction, the MDC chip receives DRAM read data on the DDB<31:0> port and loads it in its internal read data buffer. Once four bytes of data are assembled in the read data buffer, the MDC chip drives the content of the read data buffer onto the MDB<7:0> port in four separate clock cycles.

When data wrapping is selected, the order in which read data is transferred to the MIC is altered, as shown in Table 3- 1.

Table 3- 1 MDC Wrap Ordering

Cycle No.	Octaword No. Wrap Disabled (LSB D<0>=0)	Octaword No. Wrap Enabled (LSB D<0>=1)
1	1	3
2	2	4
3	3	1
4	4	2

3.2.2 Write and Wrapped Write Transactions

During a write transaction, the MDC chip receives DRAM write data on the MDB<7:0> port and loads it in its internal write data buffer in four consecutive cycles. Once the data is assembled in the write data buffer, the MDC drives the contents of the write data buffer onto the DDB<31:0> port.

When data wrapping is selected, the order in which write data is latched into the MDC is altered (see Table 3- 1) so that data written to the DRAMs is not wrapped.

Chapter 4

Memory Organization

The system supports up to seven memory modules in single- processor configurations. The physical memory composed of a single or multiple memory modules can be organized in various ways to optimize memory access.

4.1 Memory Capacity

Memory can be configured with MS7AA modules of various capacities, from 64 Mbytes to 2 Gbytes. The DRAM arrays consist of DRAMs, control signals, and address buffer components. The memory modules can use DRAM sizes of 4 Mbits or 16 Mbits. The DRAM arrays are organized into 1 to 8 strings. Each string requires 144 DRAMs (using 1M x 4 or 4M x 4 DRAMs). A single string supports 64 Mbytes of memory when configured with 4- Mbit DRAMs, and 256 Mbytes of memory when configured with 16- Mbit DRAMs. Table 1- 1 lists array capacities that can be configured based upon the number of strings on a module and the DRAM type.

Table 4- 1 Memory Array Capacity

DRAM Type (Mbits)	Number of Strings	Memory Capacity (Mbytes)
4	1	64
4	2	128
4	4	256
4	8	512
16	1	256
16	2	512
16	4	1024
16	8	2048

4.2 Memory Banking

DRAM arrays on all memory modules containing more than one string are organized as two banks, Bank 0 and Bank 1. A bank is a grouping of one or more strings that share a common address path. Each bank has its own set of control, address, and timing signals and is accessible inde-

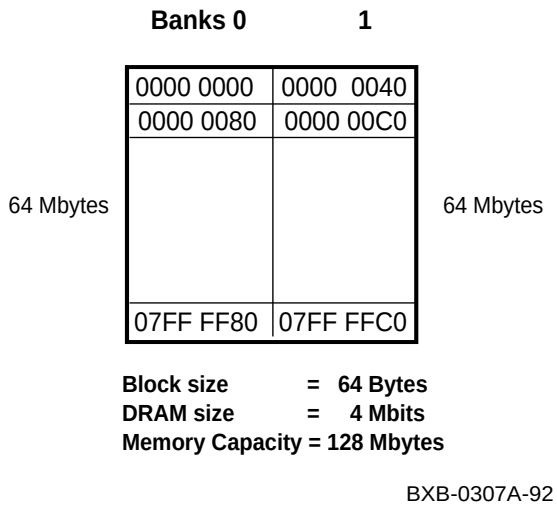
pendently. This arrangement prevents memory idling by allowing access to the second bank while the first bank is still busy.

4.3 Memory Interleaving

Further enhancement of memory performance can be achieved by interleaving the physical memory. Interleaving can be done at two levels: module and system.

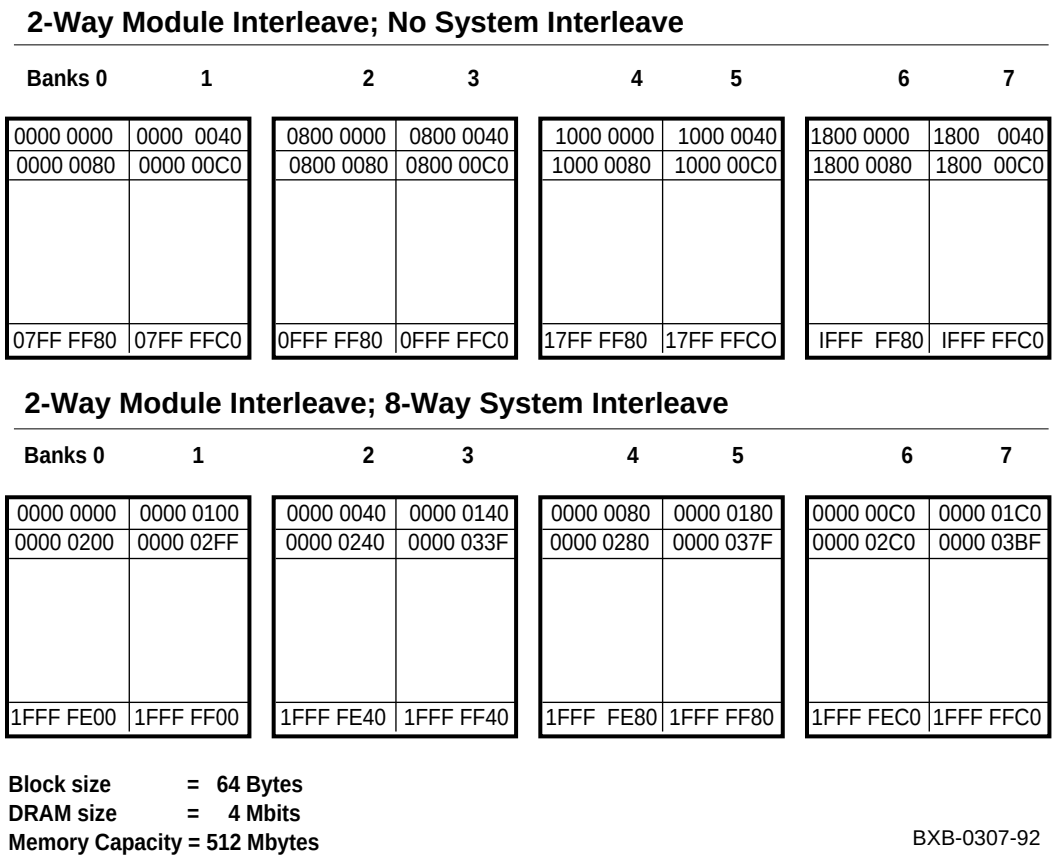
At the module level, the DRAM arrays can be interleaved on 64- byte block boundaries. Each memory module supports 2- way interleaving when configured with a minimum of two strings. Interleaving occurs between the two independently accessible banks within a module. Figure 4- 1 shows a 2- way interleaved 2- string memory module. The DRAM array in a 2- string MS7AA memory module is always interleaved.

Figure 4- 1 Two- Way Interleave of a 128- Mbyte DRAM Array



In multi- module memory subsystems, three modes of interleave are possible at the system level: default, explicit, and none. The interleave mode selection parameters are stored in the console EEPROM and can be modified through the console program. Initialization software uses registers in LSB commanders and each memory module to configure the memory interleave as specified by the EEPROM parameters. A memory configuration consisting of four memory modules supports a maximum of 8- way interleaving when each of the four modules is 2- way interleaved. The three additional modules (modules 5 to 7), if present, can be configured into the system as two modules 4- way interleaved and 1 module 2- way interleaved. Figure 4- 2 shows four memory modules in an 8- way interleaved organization.

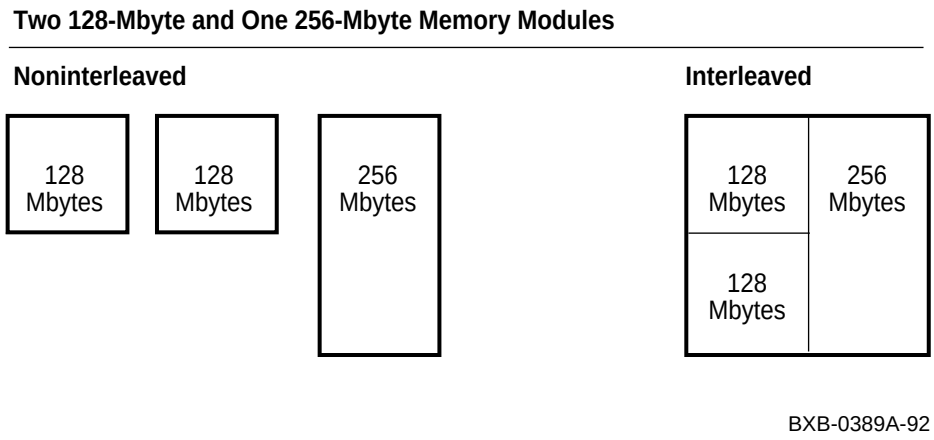
Figure 4- 2 Eight- Way System Interleave of Four 128- Mbyte Memory Modules



If the EEPROM specifies default interleave, the console attempts to form interleave sets so that the largest interleave factor is obtained for each group of DRAM arrays.

The default mode optimizes interleaving of memory in any arrangement of memory modules. For example, two 128- Mbyte modules can be combined to appear as a single 256- Mbyte module that can then be interleaved with a single 256- Mbyte module. In this configuration, three modules are 2- way interleaved among each other. This type of configuration yields 2- way module interleaving and 4- way system- level interleaving as shown in Figure 4- 3.

Figure 4- 3 Interleaving Different Size Memory Modules



If the EEPROM specifies explicit interleave sets, the console then interleaves the arrays as requested. In a noninterleave mode, the console configures arrays in order, by node number, with the lowest numbered array at the lowest physical address.

Chapter 5

LSB Memory Signals

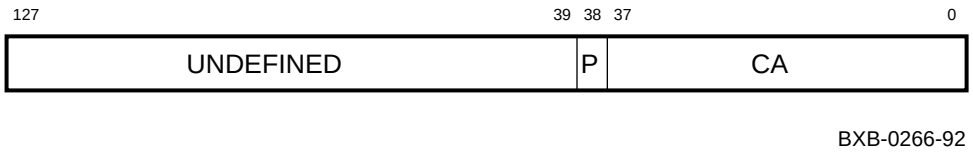
A memory module interfaces the LSB bus through a total of 173 signals. This chapter lists the LSB memory signals and describes their functions. Refer to the *Platform Technical Manual* for a complete listing of the LSB bus signals. Table 5- 1 lists the LSB memory signals and identifies their functions.

LSB Data D<127:0>

The LSB data signals transfer command/address and data information between nodes. A memory module receives command/address cycles. It drives and receives data cycles.

During command/address cycles, only D<38:0> contain valid information (see Figure 5- 1). D<38> carries the parity bit protecting the command and address information in D<37:0>. D<127:39> contain UNDEFINED data that will be ignored.

Figure 5- 1 LSB D<127:0> During Command/Address Cycles



BXB-0266-92

During CSR data cycles, D<31:0> transfer the CSR data. D<37:32> contain UNDEFINED data (see Figure 5- 2). D<38> carries the parity bit for read and write data. A memory module checks for correct parity on CSR writes and supplies correct parity on CSR reads.

NOTE: Data is ignored in D<37:32>. However, memory computes CSR data cycle parity over D<37:0>. Hence, for parity checking, D<37:32> are valid.

Figure 5- 2 LSB D<127:0> During CSR Data Cycles

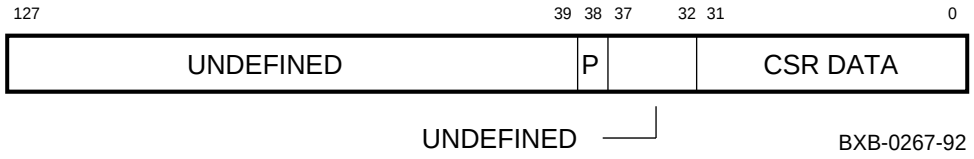


Table 5- 1 LSB Memory Signals

Signal	No. of Lines	Function
CA	1	**Command/Address.** Asserted by a bus commander to signify a command/address cycle.
CNF	1	**Confirmation.** Asserted by the selected memory module to signify acceptance of a command/address cycle.
D<127:0>	128	**Data.** Command/address and data lines.[1]
DIRTY	1	**Dirty.** Asserted by a CPU module to indicate that the targeted cache block has been modified more recently than the copy in memory. When asserted during a read transaction, the selected memory module aborts the transaction and does not return read data to the LSB bus.
ECC<27:0>	28	**Error Checking and Correction.** Error correction lines for data cycles.
ERR	1	**Error.** Asserted when a memory module detects certain LSB and memory- related errors.
NID<2:0>	3	**Node ID.** Hardwired lines that indicate the node ID (slot number) of the memory module.
PH0 (SINE)	1	**Sine.** Used to generate the eight copies of the digital clock for each module.
REQ<9:0>	10	**Arbitration Request.** Asserted by the I/O module or CPU modules to arbitrate for the LSB bus. Memory modules never assert REQ<9:0>.
RESET	1	**Reset.** Causes a system reset when asserted by a node.
STALL	1	**Stall.** Asserted when the module is unable to accept write data or supply read data.

[1] Only D<38:0> are valid during command/address cycles and LSB CSR data cycles.

LSB CA

CA is asserted by an LSB commander node when it is driving a command/address cycle on the LSB. The memory module receives but never drives this signal.

LSB CNF

CNF is asserted by memory to confirm the acceptance of a command/address cycle to either memory or node space. The selected module asserts CNF only when the LSB address matches the module's memory or node space, and no parity errors are detected.

LSB D<127:0>

Command/address and data lines.

LSB DIRTY

Memory modules monitor LSB DIRTY. If asserted during an LSB read transaction, memory does not supply read data for the transaction associated with the assertion of LSB DIRTY. The node that asserted DIRTY supplies the read data in place of memory.

LSB ECC Signals ECC<27:0>

LSB ECC signals carry the longword ECC check bits protecting D<127:0> during memory space read or write data cycles.

During the four memory space write data cycles, the selected memory module receives ECC<27:0>, with write data in D<127:0>, checks and corrects ECC errors before data is written to memory. During read data cycles from memory, a module checks and corrects single- bit ECC errors before data is driven onto LSB.

During CSR data cycles, ECC<27:0> contain UNDEFINED data that is ignored.

LSB ERR

Memory modules monitor and assert ERR under conditions discussed in the *Platform Technical Manual.* In addition, memory asserts ERR for memory- specific error conditions. Refer to Chapter 8 for additional details.

LSB NID<2:0>

These three signals indicate the node ID. They are hardwired into the backplane.

LSB PH0 (SINE)

This is the LSB- supplied clock signal. Each memory module receives and generates multiple copies of a single- phase clock.

LSB REQ<9:0>

Memory modules monitor REQ<9:0> to determine when a command/ address cycle will be driven on the LSB. The *OR* of these ten signals initiates the LSB state machines and gates the address lines to the DRAMs to help reduce power consumption. The decoded DRAM ROW address is sent to all modules and all strings independent of module selection. This is done to speed access time. If a module is not selected for the transaction, or if the bus goes to the idle state, the address drivers are held in their previous state to conserve power.

LSB RESET

All modules monitor LSB RESET. When asserted, the module is reset to its default power- up state. The assertion of LSB RESET inhibits DRAM refresh. When LSB RESET is deasserted, all memory modules start refresh operation within a specified number of cycles.

LSB STALL

Memory modules monitor and assert STALL. Memory monitors STALL to ensure that it stays in sync with the bus protocol. Memory asserts STALL for two conditions:

- A module asserts STALL when it is not able to supply read data to an LSB commander at the required time due to a memory timing conflict. STALL remains asserted until the conflict is resolved and read data can be driven on the LSB. This condition occurs when a read transaction immediately follows a write transaction on the LSB to the same memory bank. In this instance nine STALLs are asserted on the LSB.
- A module asserts STALL when it cannot supply read data or accept write data due to DRAM refresh cycles. In this case 1 to 14 STALLs are asserted on the LSB.

Chapter 6

Memory Transactions

The design of the memory module architecture ensures that the command/address issued to the DRAMs is always driven on the address lines of an idle memory bank. This speeds DRAM access during reads.

6.1 Memory Read Transactions

During a memory read transaction, DRAM read data is driven onto the MDB bus to both MIC- A and MIC- B in four consecutive clock cycles. The memory module starts the DRAM access 1.5 cycles after a command/address cycle.

A read transaction can be stalled by the LSB bus or by the memory module.

6.1.1 LSB Stalled Read Transactions

The assertion of LSB STALL by another LSB node during memory read transactions causes the memory module to delay the release of read data to the LSB bus. The DRAM read timing is not affected by the assertion of LSB STALL. Under a stall condition, the DRAM read data is latched either in the MDC chips or is transferred to the read data buffer of the MIC. The DRAM read data remains in the MIC until LSB STALL is deasserted. Read data is driven on the LSB bus two cycles after the deassertion of LSB STALL.

6.1.2 Module Stalled Read Transactions

The start of the DRAM access for a memory read transaction may be delayed due to conflicts on the memory module. These conflicts result from either DRAM refresh cycles or from DRAM write and read timing.

6.2 Memory Write Transactions

During a memory write transaction, LSB write data is latched into the MIC. Data is then transferred from the MIC to the MDC chips through the 144- bit MDB bus in four consecutive cycles. The memory module starts DRAM access a minimum of 12 bus cycles after the command/address cycle coincident with receiving the write data.

A write transaction can be stalled by the LSB bus or by the memory module.

6.2.1 LSB Stalled Write Transactions

The assertion of LSB STALL by another node delays the start of the DRAM write access until after the first LSB write data has been received. Once the DRAM write has started, it is completed in four consecutive LSB cycles.

6.2.2 Module Stalled Write Transactions

The memory module asserts LSB STALL to suppress the generation of write transactions to memory when conflicts on the memory module delay loading of write data into the DRAMs. The stall occurs on the third back-to- back write transaction to a single module preceded by a DRAM refresh. The memory module maintains LSB STALL assertion until the completion of the refresh cycle. When refresh has completed, the memory module deasserts LSB STALL, and the DRAM cycle for the write transaction starts.

6.3 Memory Transaction Ordering

In general, memory transactions are performed on a first come first served protocol. However, this protocol is overruled when a memory read follows a write transaction and targets different banks of the same memory module. In this case, the memory read transaction is always performed first to enhance system performance.

6.4 ECC Conversion

Memory modules convert LSB longword (32- bit) ECC into quadword (64-bit) ECC that is stored with LSB data on writes. During LSB reads, quadword ECC is converted to longword ECC as required by the LSB. Quadword ECC requires fewer DRAMs per string.

Chapter 7

Registers

The MS7AA memory module has two groups of registers:

- LSB required registers
- Module- specific registers

LSB required registers are used for internode communication over the LSB bus. Memory- specific registers are used to perform functions unique to the memory module.

7.1 Node Space Base Addresses

All memory registers reside on the memory controller interface. They are mapped to the node space as offsets to a base address (BB). The base address is implemented in hardware and depends on the node ID of the module, which is determined by the LSB slot occupied by the module. Table 7- 1 gives the physical base addresses of nodes on the LSB bus for processors that support a 32- bit address range (for example, KA7AA) and a 34- bit address range (for example, KN7AA).

Table 7- 1 LSB Node Space Base Addresses

Node ID (Slot)	Module	Physical Base Address (BB) (Byte) 32- Bit Range	34- Bit Range
0	CPU/Memory	F800 0000	3 F800 0000
1	CPU/Memory	F840 0000	3 F840 0000
2	CPU/Memory	F880 0000	3 F880 0000
3	CPU/Memory	F8C0 0000	3 F8C0 0000
4	CPU/Memory	F900 0000	3 F900 0000
5	CPU/Memory	F940 0000	3 F940 0000
6	CPU/Memory	F980 0000	3 F980 0000
7	CPU/Memory	F9C0 0000	3 F9C0 0000
8	I/O	FA00 0000	3 FA00 0000

7.2 Bit Access Type Acronyms

Acronyms are used throughout register descriptions to indicate the access type of the bit(s). Table 7- 2 lists those acronyms.

Table 7- 2 Access Types of Register Bits

Acronym	Access Type
RO	Read only; writes ignored.
R0	Read as zero.
R/W	Read/write.
W1C	Read/write one to clear; unaltered by a write of zero.
W1S	Write one to set; self- cleared; cannot be cleared by a write of zero.
WO	Write only.

7.3 Register Descriptions

Table 7- 3 lists the memory module registers and gives the address of each register as an offset from a selected node space base address. Functional descriptions of the memory registers follow.

Table 7- 3 Memory Module Registers

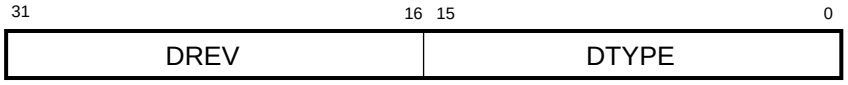
Register Name	Mnemonic	Address (Byte Offset)
LSB Required		
Device Register	LDEV	BB[1] + 0000
Bus Error Register	LBER	BB + 0040
Configuration Register	LCNR	BB + 0080
Information Base Repair Register	IBR	BB + 00C0
Error Syndrome Register 0	LBESR0	BB + 0600
Error Syndrome Register 1	LBESR1	BB + 0640
Error Syndrome Register 2	LBESR2	BB + 0680
Error Syndrome Register 3	LBESR3	BB + 06C0
Error Command Register 0	LBECR0	BB + 0700
Error Command Register 1	LBECR1	BB + 0740
Memory-Specific		
Memory Configuration Register	MCR	BB + 2000
Address MappingRegister	AMR	BB + 2040
Memory Self- Test Register 0	MSTR0	BB + 2080
Memory Self- Test Register 1	MSTR1	BB + 20C0
Failing Address Register	FADR	BB + 2100
Memory Error Register A	MERA	BB + 2140
Memory Error Syndrome Register A	MSYNDA	BB + 2180
Memory Diagnostic Register A	MDRA	BB + 21C0
Memory Check Bit Substitute Register A	MCBSA	BB + 2200
Memory Error Register B	MERB	BB + 4140
Memory Error Syndrome Register B	MSYNDB	BB + 4180
Memory Diagnostic Register B	MDRB	BB + 41C0
Memory Check Bit Substitute Register B	MCBSB	BB + 4200

[1] BB is the node space base address of the memory module in hex.

LDEV—Device Register

Address BB + 0000
Access R/W

The LDEV register is loaded during initialization with information that identifies a node.



BXB-0100-92

Table 7-4 LDEV Register Bit Definitions

Name	Bit(s)	Type	Function
DREV	<31:16>	R/W	**Device Revision.** Identifies the functional revision level of the module. The value of this field is contained in the EEPROM of the memory module and is loaded into the LDEV register by console software at system initialization. For the MS7AA memory module, the value of this field is zero.
DTYPE	<15:0>	R/W	**Device Type.** Loaded at initialization with a value of 4000 (hex), which identifies the node as a memory module. The device type may be modified by console software when the device type field is fetched from the serial EEPROM located on the memory module.

LBER—Bus Error Register

Address BB + 0040
Access R/W

The LBER register stores the error bits that are flagged in memory operations and logs the failing commander ID. The status of this register remains locked until software resets the error bit(s).

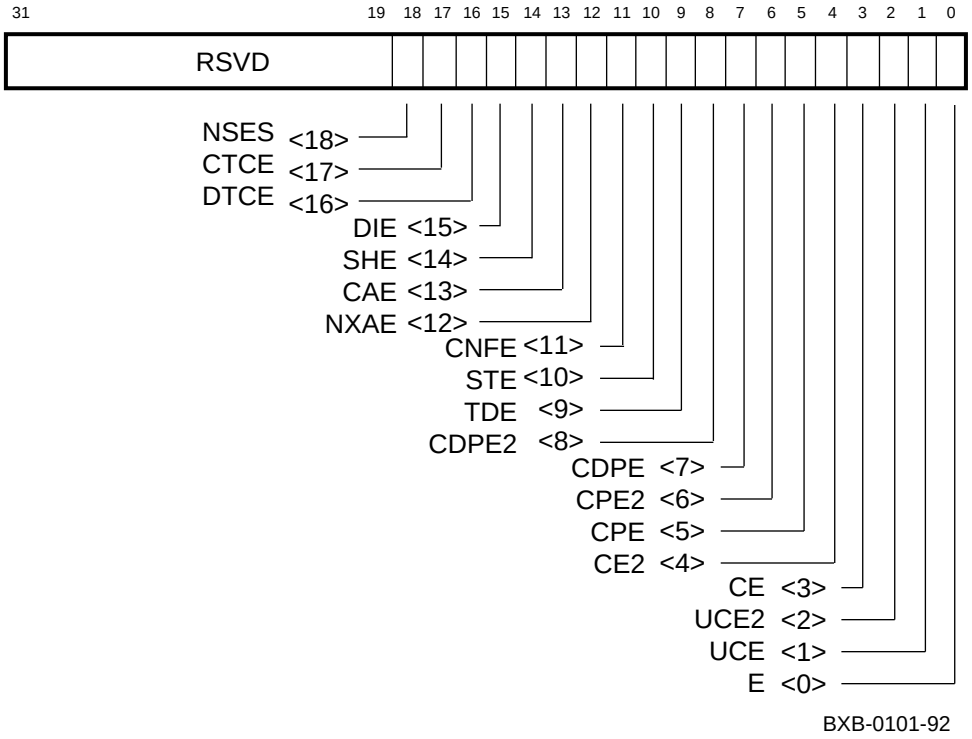


Table 7- 5 LBER Register Bit Definitions

Name	Bit(s)	Type	Function
RSVD	<31:19>	R/W, 0	**Reserved.**
NSES	<18>	RO, 0	**Node-Specific Error Summary**. Set when one of bits <11>, <10>, <9>, <5>, or <4> of the MERA register is set.
CTCE	<17>	W1C, 0	**Control Transmit Check Error.** Not implemented on memory.
DTCE	<16>	W1C, 0	**Data Transmit Check Error.** Not implemented on memory.
DIE	<15>	W1C, 0	**DIRTY Error.**
SHE	<14>	W1C, 0	**SHARED Error.** Not implemented on memory.
CAE	<13>	W1C, 0	**Command/Address Error.**
NXAE	<12>	W1C, 0	**Nonexistent Address Error.** Not implemented on memory.
CNFE	<11>	W1C, 0	**CNF Error.**
STE	<10>	W1C, 0	**STALL Error.**
TDE	<9>	W1C, 0	**Transmitter During Error.**
CDPE2	<8>	W1C, 0	**Second CSR Data Parity Error.** Not implemented on memory.
CDPE	<7>	W1C, 0	**CSR Data Parity Error.** Not implemented on memory.
CPE2	<6>	W1C, 0	**Second Command Parity Error.**
CPE	<5>	W1C, 0	**Command Parity Error.**
CE2	<4>	W1C, 0	**Second Correctable Data Error.**
CE	<3>	W1C, 0	**Correctable Data Error.**
UCE2	<2>	W1C, 0	**Second Uncorrectable Data Error.**
UCE	<1>	W1C, 0	**Uncorrectable Data Error.**
E	<0>	W1C, 0	**Error.** Signals assertion of error line.

LCNR— Configuration Register

Address BB + 0080
Access R/W

The LCNR register contains LSB systemwide configuration setup and status information.

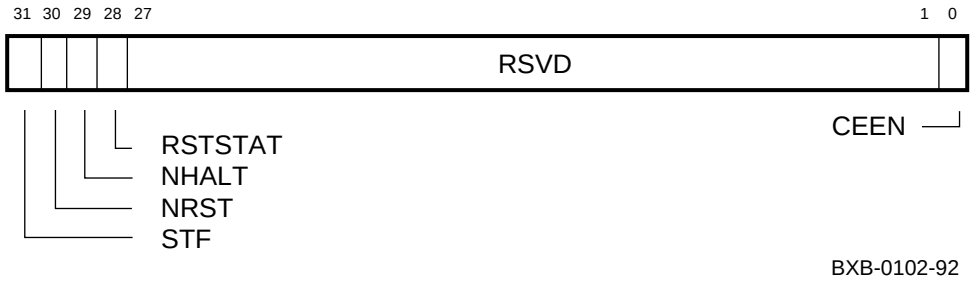


Table 7- 6 LCNR Register Bit Definitions

Name	Bit(s)	Type	Function
STF	<31>	R/W, 1	**Self-Test Fail.** When set, indicates that this node has not yet completed self- test. Memory clears this bit at the completion of self- test regardless of defective locations found in the DRAM array.
NRST	<30>	WO, 0	**Node Reset.** When set, the node enters console mode and undergoes a reset sequence. CSRs are reset to their initialized state. Any pending transactions are lost or left uncompleted. Memory self- test halts, if running, and does not restart. An internally generated reset signal remains asserted for 16 LSB bus cycles after NRST is set, providing sufficient time to reset memory state and clear NRST. If NRST is set, a value of zero must be written to the MDRA register; otherwise, the memory operation is UNDEFINED.
NHALT	<29>	R0	**Node Halt.** Not implemented on memory.
RSTSTAT	<28>	R0	**Reset Status.** Not implemented on memory.
RSVD	<27:1>	R0	**Reserved.** Read as zero.
CEEN	<0>	R/W, 0	**Correctable Error Detection Enable**. When set, enables detection of correctable errors.

IBR—Information Base Repair Register

Address BB + 00C0
Access R/W

The IBR register is used to access the EEPROM on the LSB memory and I/O modules. The EEPROM access is accomplished through continual updates of this register by software.

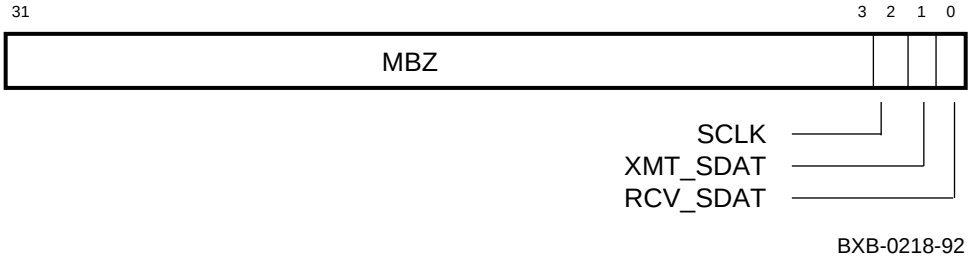


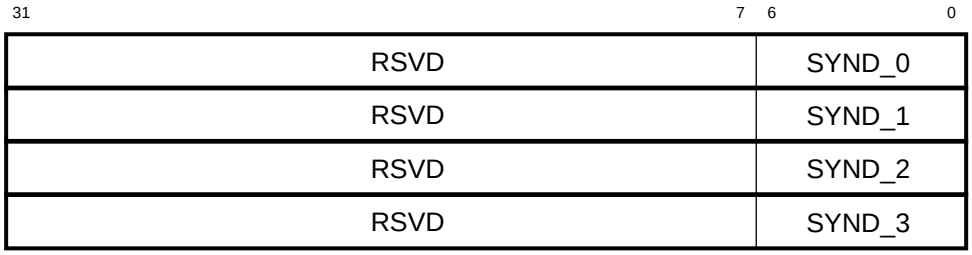
Table 7- 7 IBR Register Bit Definitions

Name	Bit(s)	Type	Function
MBZ	<31:3>	R/W, 0	**Must Be Zero.** Must always be written as zero.
SCLK	<2>	R/W, 0	**Serial Clock.** Used by software to implement the EEPROM serial clock interface. When this bit is written with a one, the EEPROM serial clock input is forced to logic high. When it is cleared, the serial clock input is forced to logic low.
XMT_SDAT	<1>	R/W, 1	**Transmit Serial Data.** Used by software to assert the serial data line of the EEPROM to either high or low logic levels. This bit is used with SCLK to transfer command/address and write data to the EEPROM.
RCV_SDAT	<0>	RO, 1	**Receive Serial Data.** Returns the status of the EEPROM serial data line. Used by software to receive serial read data and EEPROM responses. XMT_SDAT must be one to receive an EEPROM response or serial read data.

LBESR0- 3—Error Syndrome Registers

Address BB + 0600 to 06C0
Access RO

The LBESR registers contain the syndrome computed from data received on the LSB D and ECC fields during the cycle when an error was detected. These registers are locked upon the first ECC detected error and remain locked until all LSB ECC error bits in LBER are cleared.



BXB-0105-92

Table 7- 8 LBESR Register Bit Definitions

Name	Bit(s)	Type	Function
RSVD	<31:7>	R0	**Reserved.** Read as zero.
SYND_0	<6:0>	RO, 0	**Syndrome 0**. Syndrome computed from D<31:0> and ECC<6:0> during error cycle.
SYND_1	<6:0>	RO, 0	**Syndrome 1**. Syndrome computed from D<63:32> and ECC<13:7> during error cycle.
SYND_2	<6:0>	RO, 0	**Syndrome 2**. Syndrome computed from D<95:33> and ECC<20:14> during error cycle.
SYND_3	<6:0>	RO, 0	**Syndrome 3**. Syndrome computed from D<127:96> and ECC<27:21> during error cycle.

LBECR0,1—Error Command Registers

Address BB + 0700 and BB + 0740
Access RO

The LBECR registers hold the contents of the LSB command and address fields for the operation during which an error was detected.

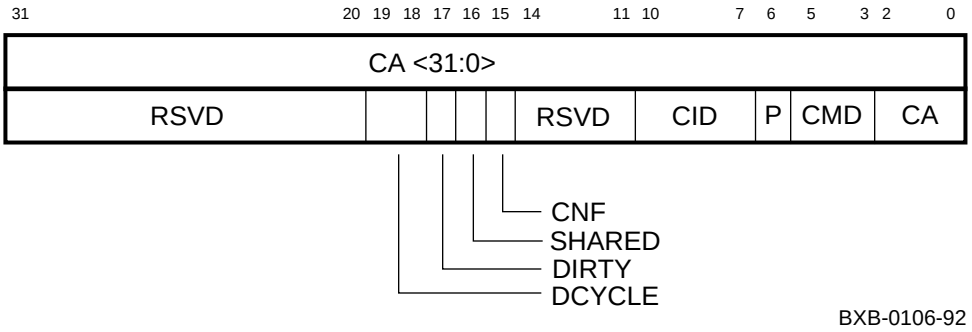


Table 7- 9 LBECR Register Bit Definitions

Name	Bit(s)	Type	Function
CA	<31:0>	RO	**Command/Address**. Contents of D<31:0> during command cycle.
RSVD	<31:20>	R0	**Reserved.** Read as zero.
DCYCLE	<19:18>	RO	**Data Cycle.** Value indicates which data cycle had the data error.

LBECR<19:18>	Data Cycle in Error
00	1
01	2
10	3
11	4

Table 7- 9 LBECR Register Bit Definitions (Continued)

Name	Bit(s)	Type	Function
DIRTY	<17>	RO	**Dirty.** Set when DIRTY is asserted for the current command.
SHARED	<16>	RO	**Shared.** Set when SHARED is asserted for the current command.
CNF	<15>	RO	**Confirmation.** Set when CNF is asserted for the current command.
RSVD	<14:11>	R0	**Reserved**. Read as zero.
CID	<10:7>	R0	**Commander ID**. Reflects the contents of REQ<3:0> during a command cycle.
P	<6>	RO	**Parity.** Contents of D<38> during command cycle.
CMD	<5:3>	RO	**Command.** Contents of D<37:35> during command cycle. CMD is decoded as follows:
CA	<2:0>	RO	**Command/ Address.** Contents of D<34:32> during command cycle.

Command	Function	DAS Value[1]
000	Read	00
001	Write	08
010	Reserved	
011	Write Victim	18
100	Read CSR	20
101	Write CSR	28
110	Reserved	
111	Private	38

[1] The hex value commonly found in the low byte of this register when less than 8 Gbytes of memory are present in the system.

MCR—Memory Configuration Register

Address BB + 2000
Access R/W

The MCR register provides information about the DRAM array. It is accessed by console software to determine module capacity, that is, the DRAM type (4-Mbit or 16-Mbit) and number of strings installed. This information is necessary to set up the eight AMR registers in each LSB commander node and the AMR register located on each memory module at BB+2040.

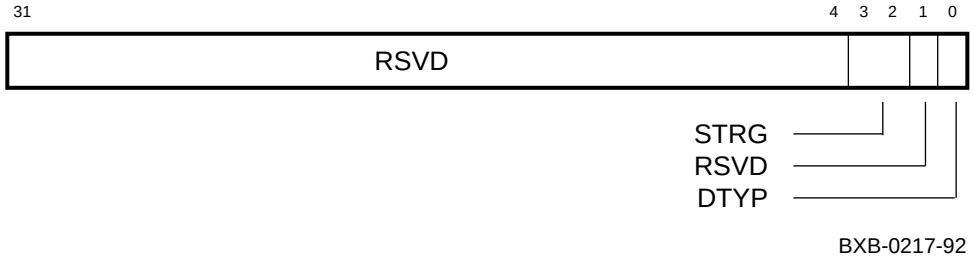


Table 7- 10 MCR Register Bit Definitions

Name	Bit(s)	Type	Function
RSVD	<31:4>	R0	**Reserved.** Read as zero.
STRG	<3:2>	RO, X	**String.** Supplies information about the number of strings installed on a module. This field, in conjunction with DTYP, indicates module capacity. It is loaded at system initialization.

MCR<3:2>	No. of Strings
00	1
01	2
10	4
11	8

RSVD	<1>	R0	**Reserved.** Reads as zero.
DTYP	<0>	RO, X	**Device Type.** Supplies information about the size of DRAM technology used for the memory module. This bit, along with STRG, indicates module capacity. It is loaded at system initialization.

MCR<0>	DRAM Type
0	4 Mbit
1	16 Mbit

AMR—Address Mapping Register

Address BB + 2040
Access R/W

The AMR register is used with the LSB memory address issued during a command/address cycle to select a memory module, a bank within a module, and adjust the address on memory modules based upon the interleave level selected, the number of strings, and the DRAM type.

This register is a single copy of the eight mapping registers that CPU modules and the IOP module must implement. When the console in the primary CPU configures memory, it determines the slots occupied by the memory modules and loads the slot number of each memory module in the AMR register of the module. Copies of the AMR registers are written into the corresponding mapping registers on all CPU nodes and the IOP node. The AMR contents for nodes 0 to 7 are placed into LMMR0- 7 registers. If a module is not present in a memory slot, the console writes a zero to bit <0> (Enable) of the corresponding LMMR register.

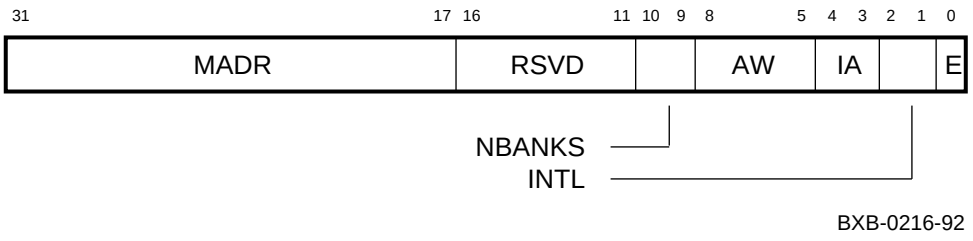


Table 7- 11 AMR Register Bit Definitions

Name	Bit(s)	Type	Function
MADR	<31:17>	R/W, 0	**Module Address.** The contents of MADR is compared with the LSB address contained on D<34:20> for a match. (MADR bit <17> correlates to LSB address line D<20>, and so on.)
RSVD	<16:11>	R0	**Reserved.** Read as zero.

Table 7- 11 AMR Register Bit Definitions (Continued)

<table>
<tr><th>Name</th><th>Bit(s)</th><th>Type</th><th>Function</th></tr>
<tr><td>NBANKS</td><td><10:9></td><td>R/W, 0</td><td>Number of Banks. Specifies the number of banks (1, 2, 4, or 8) contained on individual memory modules.
<table>
<tr><th>AMR<10:9></th><th>No. of Banks</th></tr>
<tr><td>00</td><td>1</td></tr>
<tr><td>01</td><td>2</td></tr>
<tr><td>10</td><td>Reserved</td></tr>
<tr><td>11</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>AW</td><td><8:5></td><td>R/W, 0</td><td>Address Width. Specifies the number of bits in the LSB address field <34:20> that are significant. Only those bits are compared with the same number of corrersponding bits in the AMR<MADR> (bits <31:17>). The remaining bits are ignored. The theoretical range of significant bits selected by different values of the AMR<AW> is given below. In the initialized state, no bits of D<34:20> are matched to bits AMR<31:17> (AW = 0000). When AW contains a value of 0001, only the MSBs are compared (AMR<31> is compared to LSB<34>). Access is allowed if a match occurs. If a value of F (hex) is written in AW, all bits are significant, and the LSB <34:20> must match AMR<31:17>.
<table>
<tr><th rowspan="2">AMR<8:5></th><th colspan="2">Significant Bits Compared</th></tr>
<tr><th>LSB Data</th><th>AMR<MADR></th></tr>
<tr><td>1111</td><td><34:20></td><td><31:17></td></tr>
<tr><td>1110</td><td><34:21></td><td><31:18></td></tr>
<tr><td>1101</td><td><34:22></td><td><31:19></td></tr>
<tr><td>1100</td><td><34:23></td><td><31:20></td></tr>
<tr><td>1011</td><td><34:24></td><td><31:21></td></tr>
<tr><td>1010</td><td><34:25></td><td><31:22></td></tr>
<tr><td>1001</td><td><34:26></td><td><31:23></td></tr>
<tr><td>1000</td><td><34:27></td><td><31:24></td></tr>
<tr><td>0111</td><td><34:28></td><td><31:25></td></tr>
<tr><td>0110</td><td><34:29></td><td><31:26></td></tr>
<tr><td>0101</td><td><34:30></td><td><31:27></td></tr>
<tr><td>0100</td><td><34:31></td><td><31:28></td></tr>
<tr><td>0011</td><td><34:32></td><td><31:29></td></tr>
<tr><td>0010</td><td><34:33></td><td><31:30></td></tr>
<tr><td>0001</td><td><34></td><td><31></td></tr>
<tr><td>0000</td><td>No compare</td><td></td></tr>
</table></td></tr>
</table>

Table 7- 11 AMR Register Bit Definitions (Continued)

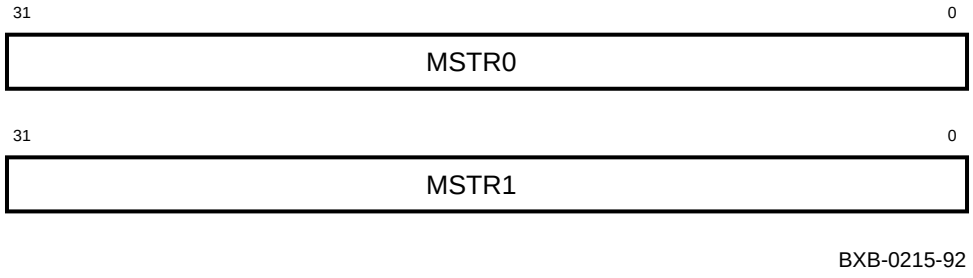
<table>
<tr><th>Name</th><th>Bit(s)</th><th>Type</th><th>Function</th></tr>
<tr><td>IA</td><td><4:3></td><td>R/W, 0</td><td>Interleave Address. Determines module selection based upon level of interleave with other memory modules.</td></tr>
<tr><td>INTL</td><td><2:1></td><td>R/W, 0</td><td>Interleave. Determines the level of interleave of this module with other modules for a given address space.

<table>
<tr><th>AMR <2:1></th><th>Interleave Level</th></tr>
<tr><td>00</td><td>1- way interleave. Module is not interleaved with any other module.</td></tr>
<tr><td>01</td><td>2- way interleave. Module is interleaved with one other module.</td></tr>
<tr><td>10</td><td>4- way interleave. Module is interleaved with three other modules.</td></tr>
<tr><td>11</td><td>Reserved.</td></tr>
</table>
</td></tr>
<tr><td>E</td><td><0></td><td>R/W, 0</td><td>Enable. Enables the module to respond to LSB memory space transactions.</td></tr>
</table>

MSTR0,1—Memory Self- Test Registers

Address BB + 2080 and BB + 20C0
Access R/W

The MSTR registers are used to isolate self- test failures to a given address segment or segments (if multiple failures) within a module. The two registers combined break up a memory module into 64 distinct address segments. Each segment resides in a range of addresses that are sized according to the selected DRAM component (4 Mbits or 16 Mbits).



BXB-0215-92

Table 7- 12 MSTR Register Bit Definitions

Name	Bit(s)	Type	Function
STFAS	<31:0>	RO, 0	**Self- Test Failing Address Segment.** A bit in the MSTR registers is set when the associated memory segment fails self- test (is defective) in a noninterleaved memory configuration. Each module executes self- test as if it were the only memory module in the system.

Table 7- 13 gives a listing of memory address segments associated with each bit in the MSTR registers. Address segments are given for 4- Mbit and 16- Mbit DRAM arrays.

Table 7- 13 Memory Error Address Segments

Failing Bit	MSTR0		MSTR1	
	4- Mbit	16- Mbit	4- Mbit	16- Mbit
0	0000 0000 to 007F FFFF	0000 0000 to 01FF FFFF	1000 0000 to 107F FFFF	4000 0000 to 41FF FFFF
1	0080 0000 to 00FF FFFF	0200 0000 to 03FF FFFF	1080 0000 to 10FF FFFF	4200 0000 to 43FF FFFF
2	0100 0000 to 017F FFFF	0400 0000 to 05FF FFFF	1100 0000 to 117F FFFF	4400 0000 to 45FF FFFF
3	0180 0000 to 01FF FFFF	0600 0000 to 07FF FFFF	1800 0000 to 11FF FFFF	4600 0000 to 47FF FFFF
4	0200 0000 to 027F FFFF	0800 0000 to 09FF FFFF	1200 0000 to 127F FFFF	4800 0000 to 49FF FFFF
5	0280 0000 to 02FF FFFF	0A00 0000 to 0BFF FFFF	1280 0000 to 12FF FFFF	4A00 0000 to 4BFF FFFF
6	0300 0000 to 037FFFFF	0C00 0000 to 0DFF FFFF	1300 0000 to 137F FFFF	4C00 0000 to 4DFF FFFF
7	0380 0000 to 08FF FFFF	0E00 0000 to 0FFF FFFF	1380 0000 to 13FF FFFF	4E00 0000 to 4FFF FFFF
8	0400 0000 to 047F FFFF	1000 0000 to 13FF FFFF	1400 0000 to 147F FFFF	5000 0000 to 51FF FFFF
9	0480 0000 to 04FF FFFF	1200 0000 to 15FF FFFF	1480 0000 to 14FF FFFF	5200 0000 to 53FF FFFF
10	0500 0000 to 057F FFFF	1400 0000 to 15FF FFFF	1500 0000 to 157F FFFF	5400 0000 to 55FF FFFF
11	0580 0000 to 05FF FFFF	1600 0000 to 17FF FFFF	1580 0000 to 15FF FFFF	5600 0000 to 57FF FFFF
12	0600 0000 to 067F FFFF	1800 0000 to 19FF FFFF	1600 0000 to 167F FFFF	5880 0000 to 59FF FFFF
13	0680 0000 to 06FF FFFF	1A00 0000 to 1BFF FFFF	1680 0000 to 16FF FFFF	5A00 0000 to 5BFF FFFF
14	0700 0000 to 077F FFFF	1C00 0000 to 1DFF FFFF	1700 0000 to 177F FFFF	5C00 0000 to 5FFF FFFF
15	0780 0000 to 07FF FFFF	1E00 0000 to 1FFF FFFF	1780 0000 to 17FF FFFF	5E00 0000 to 5FFF FFFF

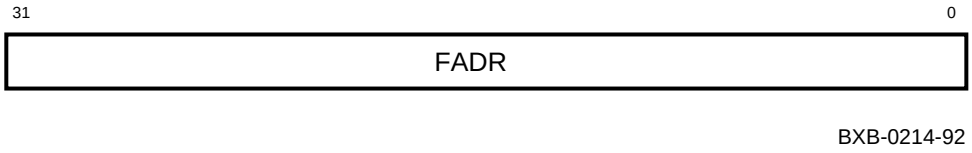
Table 7- 13 Memory Error Address Segments (Continued)

Failing Bit	MSTR0		MSTR1	
	4- Mbit	16- Mbit	4- Mbit	16- Mbit
16	0800 0000 to 087F FFFF	2000 0000 to 21FF FFFF	1800 0000 to 187F FFFF	6000 0000 to 61FF FFFF
17	0880 0000 to 08FF FFFF	2200 0000 to 23FF FFFF	1880 0000 to 18FF FFFF	6200 0000 to 63FF FFFF
18	0900 0000 to 097F FFFF	2400 0000 to 25FF FFFF	1900 0000 to 197F FFFF	6400 0000 to 65FF FFFF
19	0980 0000 to 09FF FFFF	2600 0000 to 27FF FFFF	1980 0000 to 19FF FFFF	6600 0000 to 67FF FFFF
20	0A00 0000 to 0A7F FFFF	2800 0000 to 29FF FFFF	1A00 0000 to 1A7F FFFF	6800 0000 to 69FF FFFF
21	0A80 0000 to 0AFF FFFF	2A00 0000 to 2BFF FFFF	1A80 0000 to 1AFF FFFF	6A00 0000 to 6BFF FFFF
22	0B00 0000 to 0B7F FFFF	2C00 0000 to 2DFF FFFF	1B00 0000 to 1B7F FFFF	6C00 0000 to 6DFF FFFF
23	0B80 0000 to 0BFF FFFF	2E00 0000 to 2FFF FFFF	1B80 0000 to 1BFF FFFF	6E00 0000 to 6FFF FFFF
24	0C00 0000 to 0C7F FFFF	3000 0000 to 31FF FFFF	1C00 0000 to 1C7F FFFF	7000 0000 to 71FF FFFF
25	0C80 0000 to 0CFF FFFF	3200 0000 to 33FF FFFF	1C80 0000 to 1CFF FFFF	7200 0000 to 73FF FFFF
26	0D00 0000 to 0D7F FFFF	3400 0000 to 35FF FFFF	1D00 0000 to 1D7F FFFF	7400 0000 to 75FF FFFF
27	0D80 0000 to 0DFF FFFF	3600 0000 to 37FF FFFF	1D80 0000 to 1DFF FFFF	7600 0000 to 77FF FFFF
28	0E00 0000 to 0E7F FFFF	3800 0000 to 39FF FFFF	1E00 0000 to 1E7F FFFF	7800 0000 to 79FF FFFF
29	0E80 0000 to 0EFF FFFF	3A00 0000 to 3BFF FFFF	1E80 0000 to 1EFF FFFF	7A80 0000 to 7BFF FFFF
30	0F00 0000 to 0F7F FFFF	3C00 0000 to 3DFF FFFF	1F00 to 1F7F FFFF	7C00 0000 to 7DFF FFFF
31	0F80 0000 to 0FFF FFFF	3E00 0000 to 3FFF FFFF	1F80 0000 to 1FFF FFFF	7E00 0000 to 7FFF FFFF

FADR—Failing Address Register

Address BB + 2100
Access RO

The FADR register contains the failing 32- bit LSB address any time the 64- bit ECC logic detects an ECC error during memory reads. This register is locked upon either a correctable or uncorrectable ECC error.



BXB-0214-92

Table 7- 14 FADR Register Bit Definitions

Name	Bit(s)	Type	Function
FADR	<31:0>	RO	**Failing Address.** Contains the LSB address driven on D<31:0> during a command/address cycle. The failing address bits are saved when the memory interface controller detects either a correctable or uncorrectable ECC error within the 64- bit ECC logic during LSB memory read transactions. If this field is already locked with the failing address of a correctable ECC error when an uncorrectable ECC error occurs, it is updated with the failing address of the uncorrectable error.

NOTE: The FADR register does not capture the entire LSB address. It drops address bits D<34:32>.

MERA—Memory Error Register A

Address BB + 2140
Access R/W

The MERA register provides additional error information about node- specific error conditions that are captured in the MIC- A. This register includes two bits that, when set, indicate the occurrence of an error on the MIC- B.

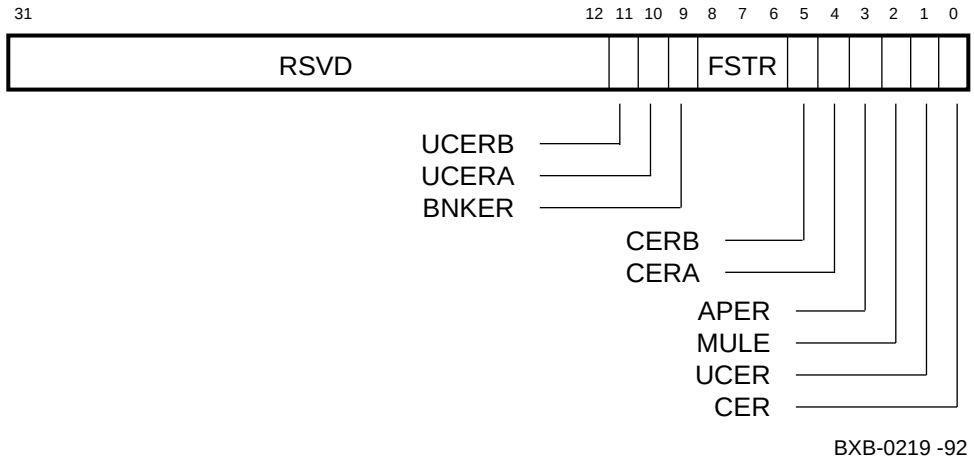


Table 7- 15 MERA Register Bit Definitions

Name	Bit(s)	Type	Function
RSVD	<31:12>	R0	**Reserved.** Read as zero.

Table 7- 15 MERA Register Bit Definitions (Continued)

Name	Bit(s)	Type	Function
UCERB	<11>	W1C, 0	**Uncontrollable Error on MIC- B.** Set when MIC-B detects an uncorrectable 64- bit ECC error. The high state of this bit locks bits <8:6> of this register and the contents of the FADR register. It also blocks UCERA (MERA<10>) from being set. If MIC- A and MIC- B were to detect simultaneous 64-bit ECC errors, then either bits <11> and <10>, or bits <5> and <4> would be set at the same time. The failing string and the address in the FADR register would pertain to failures detected in both MIC-A and MIC- B. If either MIC- A or MIC- B detects an error, then the setting of the appropriate bit and the blocking of the other bit in the combination allows isolation of independent failures on MIC- A or MIC-B to the failing string and LSB address.
UCERA	<10>	W1C, 0	**Uncorrectable Error on MIC- A.** Set when MIC-A detects an uncorrectable 64- bit ECC error. The high state of this bit locks bits <8:6> of this register and the contents of the FADR register. It also blocks UCERB (MERA<11>) from being set. See description of UCERB above.
BNKER	<9>	W1C, 0	**Bank Conflict Error**. When set, indicates that an LSB request was made to a currently active memory bank. When this bit is set, LSB ERR is asserted.
FSTR	<8:6>	RO, 0	**Failing String.** Indicates which of the eight strings were being accessed during an LSB memory read when an uncorrectable or correctable ECC error was detected by memory's 64- bit ECC logic. This field, along with the two syndrome registers, isolates single- bit errors to a failing DRAM component. FSTR is locked when a correctable or uncorrectable error bit is set in either MIC- A or MIC- B. If this field is already locked with the failing address of a correctable ECC error when an uncorrectable ECC error occurs, it is updated with the failing address of the uncorrectable error.
CERB	<5>	W1C, 0	**Correctable ECC Error on MIC- B.** Set when a correctable ECC error is detected on MIC- B. The high state of this bit locks bits <8:6> of this register and the contents of the FADR register. It also blocks CERA MERA<4>) from being set. See description of UCERB (MERA<11>) above.

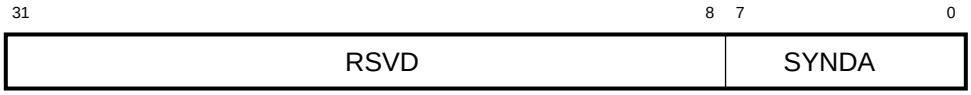
Table 7- 15 MERA Register Bit Definitions (Continued)

Name	Bit(s)	Type	Function
CERA	<4>	W1C, 0	**Correctable ECC Error on MIC- A.** Set when a correctable ECC error is detected on MIC- A. The high state of this bit locks bits <8:6> of this register, the contents of the FADR register and the contents of the MSYNDA register. It also blocks CERB (MERA<5>) from being set. See description of UCERB (MERA<11>) above.
APER	<3>	W1C, 0	**Address Parity Error.** Set when a row or column address parity error is detected. If APER is set, UCERA (MERA<1>) is also set.
MULE	<2>	W1C, 0	**Multiple Errors.** Set when a second correctable or uncorrectable error is detected after either UCER (MERA<1>) or CER (MERA<0>) has been set. This bit sets only if an error is detected on a subsequent read transaction. It does not set if multiple errors are detected within the same transaction.
UCER	<1>	W1C, 0	**Uncorrectable Read Error**. Set when an uncorrectable read data error is detected within memory's 64- bit ECC logic protecting D<63:0>. Memory supplies the data with uncorrectable errors forced on LSB check bits ECC<13:0>. When this bit sets, the Memory Error Syndrome Register A is locked with the failing syndrome.
CER	<0>	W1C, 0	**Correctable Read Error.** Set when a correctable read data error is detected within memory's 64- bit ECC logic protecting D<63:0>. Memory corrects the data in error and supplies good data to LSB. When this bit sets, the Memory Error Syndrome Register A is locked with the failing syndrome.

MSYNDA—Memory Error Syndrome Register A

Address BB + 2180
Access RO

The MSYNDA register is used to determine which bit is in error when a correctable ECC error is detected by the module during an LSB memory read transaction. This register contains the syndrome bits generated by memory's 64- bit ECC logic protecting D<63:0>. The error syndrome is decoded by software to the bit in error so that it can be logged accordingly. This register is locked when either a correctable read error bit (CER) or an uncorrectable read error bit (UCER) is set in the MERA register. MSYNDA is unlocked when the error bit or bits are cleared in the MERA register. If this register is already locked with the failing syndrome of a correctable ECC error when an uncorrectable ECC error occurs, it is updated with the uncorrectable error syndrome.



BXB-0223-92

Table 7- 16 MSYNDA Register Bit Definitions

Name	Bit(s)	Type	Function
RSVD	<31:8>	R0	**Reserved**. Read as zero.
SYNDA	<7:0>	RO, 0	**Syndrome A.** Contains the error syndrome protecting D<63:0>.

MDRA—Memory Diagnostic Register A

Address BB + 21C0
Access R/W

The MDRA register is used by diagnostics to help isolate failures and to force specific error conditions within the module and the MIC- A chip. When MDRA is written, MDRB<7:0> are written concurrently. This feature allows syncronization of diagnostic operations between MIC- A and MIC- B. Bits<7:0> of MDRA and MDRB registers have one- to- one matching functions.

NOTE: This register is reserved for Digital use.

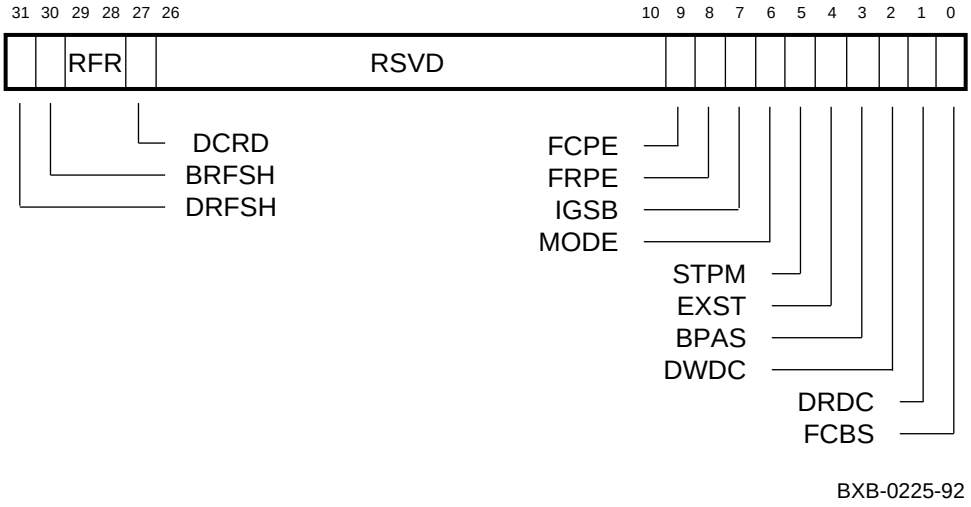


Table 7- 17 MDRA Register Bit Definitions

Name	Bit(s)	Type	Function
DRFSH	<31>	R/W, 0	**Disable Refresh.** If set, "on- board" refresh of the module is disabled and diagnostic burst refresh (see BRFSH, MDRA<30>) is enabled. When this bit is set concurrently with BRFSH, a burst refresh cycle is executed.

Table 7- 17 MDRA Register Bit Definitions (Continued)

Name	Bit(s)	Type	Function
BRFSH	<30>	WO, 0	**Burst Refresh**. When BRFSH is set, and DRFSH (MDRA<31>) is also set, a single row address within the accessed DRAMs is refreshed as per CAS before RAS refresh operation. When this bit is set concurrently with DRFSH, a burst refresh cycle is executed.
RFR	<29:28>	R/W, 0	**Refresh Rate**. Determines the refresh rate of the module.
DCRD	<27>	R/W, 0	**Disable CRD Assertion.** If set, LSB_CRD does not assert when memory detects a read CRD error in the 64- bit ECC logic.
RSVD	<26:10>	R0	**Reserved**. Read as zero.
FCPE	<9>	R/W, 0	**Force Column Parity Error**. If set, incorrect column address parity is written into the addressed location in memory.
FRPE	<8>	R/W, 0	**Force Row Parity Error**. If set, incorrect row address parity is written into the addressed location in memory.
IGSB	<7>	R/W, 0	**Ignore Single- Bit Self- Test Failures**. If set, self-test logic does not report errors in the MSTR0/ MSTR1 registers when a single bit is detected to be in error for each octaword read. Only errors of two or more bits are reported.
MODE	<6>	R/W, 0	**Self- Test Mode**. If set, self- test executes a defined "ones/zeros" data pattern when running under diagnostic control. Two passes are executed. In the first pass, all data bits in an odd hexword address are written with ones, while those in an even hexword are written with zeros. In the second pass, hexword addresses are written with inverse data patterns.

MDRA<29:28>	Refresh Rate
00	1 x default
01	2 x default
10	4 x default
11	Reserved

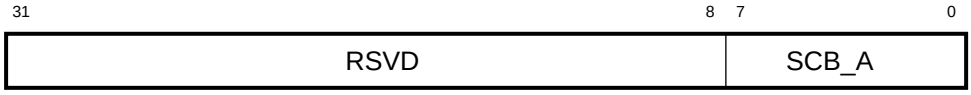
Table 7- 17 MDRA Register Bit Definitions (Continued)

Name	Bit(s)	Type	Function
STPM	<5>	R/W, 0	**Self- Test Pause Mode.** If set, self- test completes one of its three passes. To execute the next pass, EXST (MDRA<4>) must be set. This feature is necessary for memory manufacturing tests. STPM must be set at the start of each pass. Otherwise, self- test begins at pass 1.
EXST	<4>	R/W, 1	**Execute Self- Test**. If EXST is set, memory self- test is invoked. The self- test logic examines this bit to determine if self- test should be executed. This bit is set upon system power- up or LSB reset. It is cleared when self- test completes execution. It is also cleared in diagnostic mode when STPM is set for each pass that the self- test state machine completes.
BPAS	<3>	R/W, 0	**Bypas**s. If set, both the read data path and write data path ECC logic is bypassed to aid in memory manufacturing tests. LSB ECC<7:0> is directly mapped to the eight ECC check bits protecting D<63:0> in the MIC- A.
DWDC	<2>	R/W, 0	**Disable Write Data Correctio**n. If set, ECC correction of LSB memory write data is disabled. Uncorrected data is written to memory with a no error ECC check bit code. The setting of this bit affects the correction of D<63:0>. Thirty- two bit ECC errors, if present, are logged.
DRDC	<1>	R/W, 0	**Disable Read Data Correction.** If set, ECC correction of memory read data is disabled. The setting of this bit affects the correction of D<63:0>.
FCBS	<0>	R/W, 0	**Force Check Bit Substitution.** If set, the contents of the MCBSB register is gated into the 64- bit write ECC logic in place of the DRAM check bits. The setting of this bit affects D<63:0>.

MCBSA—Memory Check Bit Substitute Register A

Address BB + 2200
Access R/W

The MCBSA register is used to check the 64- bit ECC logic on the memory module during memory read transactions. This register is first loaded by diagnostics with substitute bits for DRAM check bits. The contents of the MCBSA register is then written to a selected memory location. When the write transaction is complete, each of the four octawords contains the substituted check bits and data issued on the bus. The location in memory written with substitute check bits will contain an UNDEFINED ECC code. Software executing such tests must ensure that memory is initialized to good ECC by reexecuting self- test or by other means.



BXB-0221-92

Table 7- 18 MCBSA Register Bit Definitions

Name	Bit(s)	Type	Function
RSVD	<31:8>	R0	**Reserved.** Read as zero.
SCB_A	<7:0>	R/W, 0	**Substituted Check Bits A.** SCB_A is used in place of the DRAM check bits protecting D<63:0> when DRDC (MDRA <1>) is set.

MERB—Memory Error Register B

Address BB + 4140
Access R/W

The MERB register provides additional error information about node- specific error conditions that are captured in the MIC- B chip

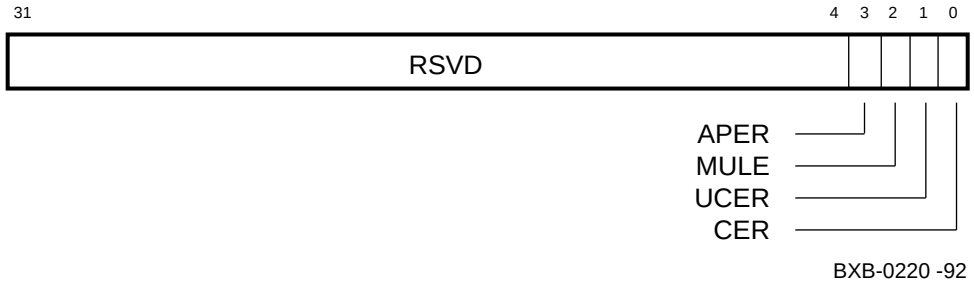


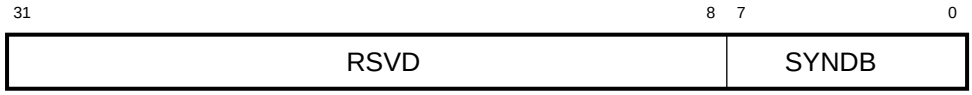
Table 7- 19 MERB Register Bit Definitions

Name	Bit(s)	Type	Function
RSVD	<31:4>	R0	**Reserved.** Read as zero.
APER	<3>	W1C, 0	**Address Parity Error.** Set when a row or column address parity error is detected. If APER is set, UCER (MERB<1>) is also set.
MULE	<2>	W1C, 0	**Multiple Errors.** Set when a second correctable or uncorrectable error is detected after either UCER (MERB<1>) or CER (MERB<0>) has been set. This bit sets only if an error is detected on a subsequent read transaction. It does not set if multiple errors are detected within the same transaction.
UCER	<1>	W1C, 0	**Uncorrectable Read Error**. Set when an uncorrectable read data error is detected within memory's 64- bit ECC logic protecting D<127:64>. Memory supplies the data with uncorrectable errors forced on LSB check bits ECC<27:14>. When this bit sets, the Memory Error Syndrome Register A is locked with the failing syndrome.
CER	<0>	W1C, 0	**Correctable Read Error.** Set when a correctable read data error is detected within memory's 64- bit ECC logic protecting D<127:64>. Memory corrects the data in error and supplies good data to the LSB. When this bit sets, the Memory Error Syndrome Register B is locked with the failing syndrome.

MSYNDB—Memory Error Syndrome Register B

Address BB + 4180
Access RO

The MSYNDB register is used to determine which bit is in error when a correctable ECC error is detected by the module during an LSB memory read transaction. This register contains the syndrome bits generated by memory's 64- bit ECC logic protecting D<63:0>. The error syndrome is decoded by software to the bit in error so that it can be logged accordingly. This register is locked when either a correctable read error bit (CER) or an uncorrectable read error bit (UCER) is set in the MERB register. MSYNDB is unlocked when the error bit or bits are cleared in the MERB register. If this register is already locked with the failing syndrome of a correctable ECC error when an uncorrectable ECC error occurs, it is updated with the uncorrectable error syndrome.



BXB-0224-92

Table 7- 20 MSYNDB Register Bit Definitions

Name	Bit(s)	Type	Function
RSVD	<31:8>	R0	**Reserved**. Read as zero.
SYNDB	<7:0>	RO, 0	**Syndrome B.** Contains the error syndrome protecting D<127:64>.

MDRB—Memory Diagnostic Register B

Address BB + 41C0
Access R/W

The MDRB register is used by diagnostics to help isolate failures and to force specific error conditions within the module and the MIC- B chip. When the MDRA register is written, MDRB<7:0> are written concurrently. This feature allows syncronization of diagnostic operations between MIC- A and MIC- B. Bits<7:0> of MDRA and MDRB registers have one to one matching functions.

NOTE: This register is reserved for Digital use.

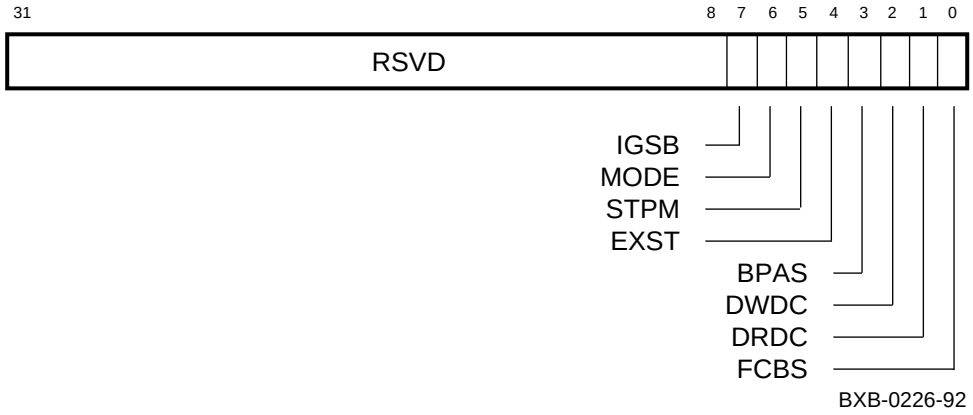


Table 7- 21 MDRB Register Bit Definitions

Name	Bit(s)	Type	Function
RSVD	<31:8>	R0	**Reserved.** Read as zero.
IGSB	<7>	R/W, 0	**Ignore Single- Bit Self- Test Failures**. If set, self-test logic does not report errors in the MSTR0/ MSTR1 registers when a single bit is detected to be in error for each octaword read. Only errors of two or more bits are reported.

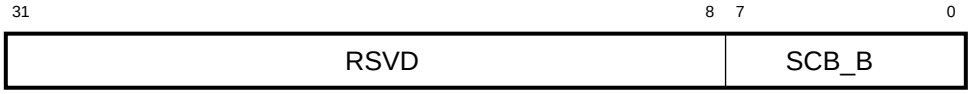
Table 7- 21 MDRB Register Bit Definitions (Continued)

Name	Bit(s)	Type	Function
MODE	<6>	R/W, 0	**Self- Test Mode**. If set, self- test executes a defined "ones/zeros" data pattern when running under diagnostic control. Two passes are executed. In the first pass, all data bits in an odd hexword address are written with ones, while those in an even hexword are written with zeros. In the second pass, hexword addresses are written with inverse data patterns.
STPM	<5>	R/W, 0	**Self- Test Pause Mode.** If set, self- test completes one of its three passes. To execute the next pass, EXST (MDRB<4>) must be set. This feature is necessary for memory manufacturing tests. STPM must be set at the start of each pass. Otherwise, self- test begins at pass 1.
EXST	<4>	R/W, 0	**Execute Self- Test**. If EXST is set, memory self-test is invoked. The self- test logic examines this bit to determine if self- test should be executed. This bit is set upon system power- up or LSB reset. It is cleared when self- test completes execution. It is also cleared in diagnostic mode when STPM is set for each pass that self- test completes.
BPAS	<3>	R/W, 0	**Bypas**s. If set, both the read data path and write data path ECC logic is bypassed to aid in memory manufacturing tests. LSB ECC<15:8> is directly mapped to the eight ECC check bits protecting D<127:64> in the MIC- B chip.
DWDC	<2>	R/W, 0	**Disable Write Data Correctio**n. If set, ECC correction of LSB memory write data is disabled. Uncorrected data is written to memory with a no error ECC check bit code. The setting of this bit affects the correction of D<127:64>. Thirty- two bit ECC errors, if present, are logged.
DRDC	<1>	R/W, 0	**Disable Read Data Correction.** If set, ECC correction of memory read data is disabled. The setting of this bit affects the correction of D<127:64>.
FCBS	<0>	R/W, 0	**Force Check Bit Substitution.** If set, the contents of the MCBSB register is gated into the 64- bit write ECC logic in place of the DRAM check bits. The setting of this bit affects D<127:64>.

MCBSB—Memory Check Bit Substitute Register B

Address BB + 2200
Access R/W

The MCBSB register is used to check the 64- bit ECC logic on the memory module during memory read transactions. This register is first loaded by diagnostics with substitute bits for DRAM check bits. The contents of the MCBSB register is then written to a selected memory location. When the write transaction is complete, each of the four octawords contains the substituted check bits and data issued on the bus. The location in memory written with substitute check bits will contain an UNDEFINED ECC code. Software executing such tests must ensure that memory is initialized to good ECC by reexecuting self- test or by other means.



BXB-0222-92

Table 7- 22 MCBSB Register Bit Definitions

Name	Bit(s)	Type	Function
RSVD	<31:8>	R/W, 0	**Reserved.** Read as zero.
SCB_B	<7:0>	R/W, 0	**Substituted Check Bits B.** SCB_B is used in place of the DRAM check bits protecting D<127:64> when DRDC (MDRB <1>) is set.

Chapter 8

Error Conditions

The memory module detects two categories of ECC errors:

- Memory- related LSB errors
- Memory- specific errors

This chapter describes the two types of errors and explains how they are reported in error syndrome registers.

8.1 Memory- Related LSB Errors

In general, memory responds as follows to LSB errors it detects during memory read or write transactions:

- **Command/Address Cycles**
  Memory reports parity errors that it detects during command/address cycles and captures the contents of D<38:0> in the LBECR0 and LBECR1 registers. It does not act on the command or the address.
- **CSR Write Data Cycles**
  Memory does not write CSR data to the specified memory CSR when a parity error is detected. Memory asserts LSB ERR.
- **Memory Write Data Cycles**
  If a single- bit ECC error is detected on memory write data, the bit in error is corrected before data is written to memory. If an uncorrectable ECC error is detected on memory write data, no correction is made to the data. The bad data is written to memory along with an uncorrectable ECC code forced on the 64- bit ECC check bits.
- **Memory Read Data Cycles**
  See memory- specific ECC errors in the MERA and MERB registers.

8.2 Memory- Specific Errors

Memory- specific errors relate to the 64- bit ECC logic that protects the DRAM array. These errors are reported in Memory Error Register A (MERA) and Memory Error Register B (MERB). Memory- specific errors are detected only during LSB memory reads.

8.3 Error Syndrome Decoding

On the LSB bus, data is protected by 32- bit ECC error logic. However, on the memory module, error logic implements a 64- bit ECC. Therefore, conversion of quadword (64- bit) ECC to longword (32- bit) ECC must be per-

formed on memory read data. Conversely, longword to quadword ECC is performed on memory write data. The conversion logic is implemented on the MIC.

8.3.1 Quadword Memory Error Syndrome Decoding

When an ECC error is detected by the 64- bit memory ECC logic, the error syndrome registers MSYNDA/MSYNDB are locked. Each of these registers contains the error syndrome associated with a failure in either the MERA register or the MERB register.

If a correctable or uncorrectable error bit is set, the error syndrome register will contain the failing syndrome. If the error is correctable, logging software can use this information to determine the bit in error and log it appropriately. This information is of little use, however, if the error is uncorrectable.

Table 8- 1 shows the 64- bit ECC codes used, with associated check bits for each of the 64 data bits. Error logging software can use this table to determine the bit in error whenever single- bit errors are detected.

Table 8- 1 Quadword Error Syndrome Table

MIC- A Bit in Error	MIC- B Bit in Error	Syndrome Value (Hex)
No Error	No Error	00
0	64	13
1	65	23
2	66	43
3	67	83
4	68	2F
5	69	F1
6	70	0D
7	71	07
8	72	D0
9	73	70
10	74	4F

Table 8- 1 Quadword Error Syndrome Table (Continued)

MIC- A Bit in Error	MIC- B Bit in Error	Syndrome Value (Hex)
11	75	F8
12	76	61
13	77	62
14	78	64
15	79	68
16	80	1C
17	81	2C
18	82	4C
19	83	8C
20	84	15
21	85	25
22	86	45
23	87	85
24	88	19
25	89	29
26	90	49
27	91	89
28	92	1A
29	93	2A
30	94	4A
31	95	8A
32	96	51
33	97	52
34	98	54
35	99	58
36	100	91
37	101	92
38	102	94
39	103	98
40	104	A1
41	105	A2
42	106	A4
43	107	A8
44	108	31
45	109	32
46	110	34
47	111	38
48	112	16
49	113	26
50	114	46
51	115	89
52	116	1F
53	117	F2
54	118	0B
55	119	0E

Table 8- 1 Quadword Error Syndrome Table (Continued)

MIC- A Bit in Error	MIC- B Bit in Error	Syndrome Value (Hex)
56	120	B0
57	121	E0
58	122	8F
59	123	F4
60	124	C1
61	125	C2
62	126	C4
63	127	C8
CB0	CB8	01
CB1	CB9	02
CB2	CB10	04
CB3	CB11	08
CB4	CB12	10
CB5	CB13	20
CB6	CB14	40
CB7	CB15	80
Row Parity	FD	
Column Parity	FE	

8.3.2 Longword Memory Error Syndrome Decoding

When an ECC error is detected by the LSB ECC logic, the error syndrome registers LBESR0–3 are locked. Each of these registers contains the error syndrome associated with a failure as seen on the LSB bus.

If a correctable or uncorrectable error bit is set, the error syndrome register will contain the failing syndrome. If the error is correctable, logging software can use this information to determine the bit in error and log it appropriately. This information is of little use, however, if the error is uncorrectable.

Table 8- 2 shows the 32- bit ECC codes used, with associated check bits for each of the 32 data bits. Error logging software can use this table to determine the bit in error whenever single- bit errors are detected.

Table 8- 2 Longword LSB Error Syndrome Table

MIC- A Bit in Error	MIC- B Bit in Error	Syndrome Value (Hex)
No Error	No Error	00
0,32	64,96	4F
1,33	65,97	4A
2,34	66,98	52
3,35	67,99	54
4,36	68,100	57
5,37	69,101	58
6,38	70,102	5B
7,39	71,103	5D
8,40	72,104	23
9,41	73,105	25
10,42	74,106	26
11,43	75,107	29
12,44	76,108	2A
13,45	77,109	2C
14,46	78,110	31
15,47	79,111	34
16,48	80,112	0E
17,49	81,113	0B
18,50	82,114	13
19,51	83,115	15
20,52	84,116	16
21,53	85,117	19
22,54	86,118	1A
23,55	87,119	1C
24,56	88,120	62
25,57	89,121	64
26,58	90,122	67
27,59	91,123	68
28,60	92,124	6B
29,61	93,125	6D
30,62	94,126	70
31,63	95,127	75
CB0,CB7	CB14,CB21	01
CB1,CB8	CB15,CB22	02
CB2,CB9	CB16,CB23	04
CB3,CB10	CB17,CB24	08
CB4,CB11	CB18,CB25	10
CB5,CB12	CB19,CB26	20
CB6,CB13	CB20,CB27	40
Cycle- 0	7A	
Cycle- 1	79	

Chapter 9

Self- Test

The memory self- test performs a quick testing of the DRAM array and records sections of the array that contain defective locations. This occurs automatically at system power- up upon assertion of LSB RESET. Self-test may be invoked by setting EXST (MDRA/MDRB<4>) through a console deposit command.

9.1 Self- Test Report

When the memory module starts self- test, it sets STF (LCNR<31>). The memory self- test does not provide a pass/fail status. It flags the defective segments in memory by setting associated bits in the MSTR0 and MSTR1 registers (see Table 7- 13) and clears EXST and STF when it is done. Clearing of STF lights the module LED, indicating the completion of the test procedures.

Address segments are mapped according to the module's total possible capacity (maximum of 8 strings), not to the actual memory capacity implemented on the module. For example, the maximum possible capacity of a memory module using 4- Mbit DRAMs is 512 Mbytes (2 Gbytes for 16-Mbit DRAMs). In this configuration, each bit in the MSTR0 and MSTR1 registers is associated with an 8- Mbyte segment of memory (32 Mbytes for 16- Mbit DRAMs). Bit zero of MSTR0 describes an address segment from 0 to 8 Mbytes (0 to 32 Mbytes for 16- Mbit DRAMs); bit one describes an address segment from 8 to 16 Mbytes (32 to 64 Mbytes for 16- bit DRAMs), and so on.

9.2 Self- Test Modes

Self- test can be conducted in several different operating modes selected by configuring the MDRA and MDRB registers. One mode of operation causes the self- test logic to ignore all single- bit errors. When this mode is selected, single- bit failures do not set any of the 64 address segment bits in MSTR0 or MSTR1. However, if a multiple- bit error is detected in a memory segment, the associated address segment is set. By default, single- bit errors are recorded in MSTR0 and MSTR1.

9.3 Self- Test Performance

The duration of the memory self- test is affected by the following parameters:

- Memory array capacity
- Memory array architecture
- DRAM size
- DRAM speed

The capacity of the memory module is the main factor that determines the length of self- test. Table 9- 1 shows self- test times for memory modules of different capacities.

Table 9- 1 Memory Self- Test Times

Module Capacity (Mbytes)	Self- Test Duration (Seconds) 4- Mbit DRAM	16- Mbit DRAM
64	1.3	N/A[1]
128	1.3	N/A
256	2.7	5.4
512	5.4	5.4
1024	N/A	10.7
2028	N/A	21.4

[1] Not applicable.

Index

E

F

I

L

M

N

O

P

Q

R

S

T

U

W

X