# JEDEC
# STANDARD

**DDR5 Serial Presence Detect (SPD) Contents**

**JESD400-5**

**Version 1.0**

**June 2022**

**JEDEC SOLID STATE TECHNOLOGY ASSOCIATION**

NOTICE

JEDEC standards and publications contain material that has been prepared, reviewed, and approved through the JEDEC Board of Directors level and subsequently reviewed and approved by the JEDEC legal counsel.

JEDEC standards and publications are designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvement of products, and assisting the purchaser in selecting and obtaining with minimum delay the proper product for use by those other than JEDEC members, whether the standard is to be used either domestically or internationally.

JEDEC standards and publications are adopted without regard to whether or not their adoption may involve patents or articles, materials, or processes. By such action JEDEC does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the JEDEC standards or publications.

The information included in JEDEC standards and publications represents a sound approach to product specification and application, principally from the solid state device manufacturer viewpoint. Within the JEDEC organization there are procedures whereby a JEDEC standard or publication may be further processed and ultimately become an ANSI standard.

No claims to be in conformance with this standard may be made unless all requirements stated in the standard are met.

Inquiries, comments, and suggestions relative to the content of this JEDEC standard or publication should be addressed to JEDEC at the address below, or refer to www.jedec.org under Standards and Documents for alternative contact information.

Published by
©JEDEC Solid State Technology Association 2022
3103 North 10th Street
Suite 240 South
Arlington, VA 22201-2108

**PRICE: Contact JEDEC**

Printed in the U.S.A.
All rights reserved

PLEASE!

DON'T VIOLATE
THE
LAW!

This document is copyrighted by JEDEC and may not be reproduced
without permission.

For information, contact:

JEDEC Solid State Technology Association
3103 North 10th Street
Suite 240 South
Arlington, VA 22201-2107

or refer to www.jedec.org under Standards-Documents/Copyright Information.

This page intentionally left blank

# DDR5 SERIAL PRESENCE DETECT (SPD) CONTENTS

## Contents

**Contents (cont'd)**

**Contents (cont'd)**

## Contents (cont'd)

**Contents (cont'd)**

**Tables**

-vi-

This page intentionally left blank

## DDR5 SERIAL PRESENCE DETECT (SPD) CONTENTS

From JEDEC Board Ballot JCB-21-28, formulated under the cognizance of JC-45, DRAM modules, item 2260-01M. This standard is broken down into various clauses with their respective revision numbers below.

- DDR5 SPD Document Release 1.0
- Base SPD revision 1.0
- Soldered down annex revision 1.0
- UDIMM annex revision 1.0
- RDIMM/LRDIMM annex revision 1.0
- DDIMM annex revision 1.0
- NVDIMM-N annex revision 0.1
- NVDIMM-P annex revision 1.0

## 1    Scope

This publication describes the serial presence detect (SPD) values for all DDR5 memory modules. In this context, "modules" applies to memory modules like traditional Dual In-line Memory Modules (DIMMs) or solder-down motherboard applications. The SPD data provides critical information about all modules on the memory channel and is intended to be use by the system's BIOS in order to properly initialize and optimize the system memory channels. The storage capacity of the SPD non-volatile memory is limited, so a number of techniques are employed to optimize the use of these bytes, including overlays and run length limited coding.

All unused entries will be coded as 0x00. All unused bits in defined bytes will be coded as 0 except where noted.

Timing parameters in the SPD represent the operation of the module including all DRAMs and support devices at the lowest supported supply voltages (see SPD bytes 16 through 18), and are valid from $t_{CKAVG}$min to $t_{CKAVG}$max (see SPD bytes 20 through 23).

To allow for maximum flexibility as devices evolve, SPD fields described in this document may support device configuration and timing options that are not included in the JEDEC DDR5 SDRAM data sheet (JESD79-5). Please refer to DRAM supplier data sheets or JESD79-5 to determine the compatibility of components.

## 2    History

Computer main memory buses have traditionally been defined by the generation of memory attached to the bus, e.g., EDO, SDRAM, DDR1, etc. The bus interface protocol and characteristics have largely been defined by the memory type. Clock frequency, CAS latencies, refresh recovery times and similar parameters defined the timing of signals between memory controller and the memory, and parameters such as number of ranks installed and device widths allowed system software to determine the memory capacity and similar high level characteristics of each module.

Over time, the memory bus has been extended to include additional features for application specific requirements. Registered DIMMs, for example, increased total capacity by buffering the loading of the address bus signals, allowing more DRAM to be installed. Similarly, Load Reduced DIMMs buffered the data bus as well, allowing even more ranks of memory to be supported. As each new extension to the function of the memory bus was introduced, system software combined knowledge of those extensions with information programmed into the non-volatile memory in the SPD to determine how to use and optimize the new features. Using the RDIMM as an example, systems understood than an additional clock of latency needed to be added to the DRAM latency to accommodate propagation delay through the register.

In later generations, the DRAM to host interface is completely virtualized. A memory module may have no DRAM at all, yet may use the DRAM bus to communicate with the host by emulating the DRAM channel interface. These virtual interfaces must appear to the system as one of the base module types, i.e., UDIMM, RDIMM, LRDIMM, or DDIMM. Modules that incorporate at least one non-DRAM media type for the purpose of main memory data storage are called "hybrid", they act like a DRAM but on the other side of the interface protocol are some other memory type(s).

# 3    SPD Architecture

The SPD contents architecture must support the many variations of module types while remaining efficient. A system of overlay information selected through the use of "key bytes", or selectors for the type of information to load has been implemented. The following DDR5 module SPD address map describes where the individual lookup table entries will be held in the serial non-volatile memory.

Consistent with the definition of DDR5 generation SPD devices (SPD5118) which have 16 individual write protection blocks of 64 bytes in length each, the SPD contents are aligned with these blocks as follows:

| Block | Range | | Description |
|---|---|---|---|
| 0 | 0~63 | 0x000~0x03F | Base Configuration and DRAM Parameters |
| 1 | 64~127 | 0x040~0x07F | Base Configuration and DRAM Parameters |
| 2 | 128~191 | 0x080~0x0BF | Reserved for future use |
| 3 | 192~239 | 0x0C0~0x0EF | Common Module Parameters -- See annex A.0 for details |
|   | 240~255 | 0x0D0~0x0FF | Standard Module Parameters -- See annexes A.x for details |
| 4 | 256~319 | 0x100~0x13F | Standard Module Parameters -- See annexes A.x for details |
| 5 | 320~383 | 0x140~0x17F | Standard Module Parameters -- See annexes A.x for details |
| 6 | 384~447 | 0x180~0x1BF | Standard Module Parameters -- See annexes A.x for details |
| 7 | 448~509 | 0x1C0~0x1FD | Reserved for future use |
|   | 510~511 | 0x1FE~0x1FF | CRC for SPD bytes 0~509 |
| 8 | 512~575 | 0x200~0x23F | Manufacturing information |
| 9 | 576~639 | 0x240~0x27F | Manufacturing information |
| 10 | 640~703 | 0x280~0x2BF | End User Programmable |
| 11 | 704~767 | 0x2C0~0x2FF | End User Programmable |
| 12 | 768~831 | 0x300~0x33F | End User Programmable |
| 13 | 832~895 | 0x340~0x37F | End User Programmable |
| 14 | 896~959 | 0x380~0x3BF | End User Programmable |
| 15 | 960~1023 | 0x3C0~0x3FF | End User Programmable |

Operating parameters for the different module types are defined in the following Annexes and will reside in the appropriate address ranges of the SPD address map depending on the module type. Please see Overlay Schema for further detail.

- Annex A.0: Common SPD Bytes for All Module Types
- Annex A.1: Solder down memory applications
- Annex A.2: UDIMMs
- Annex A.3: RDIMM and LRDIMM
- Annex A.4: Reserved
- Annex A.5: DDIMM
- Annex A.6: NVDIMM-N
- Annex A.7: NVDIMM-P

# 4    Overlay Schema

The following Schema exemplifies the manner in which the base configuration information along with the Annexes are to be overlaid onto the appropriate address spaces in order to provide a complete definition of the module.

## 4.1    Solder Down Overlay Schema

Key Byte 3 contains any of the following values:
- 0x0B, Solder Down

| Block | Range | | Description |
|---|---|---|---|
| 0 | 0~63 | 0x000~0x03F | Base Configuration and DRAM Parameters |
| 1 | 64~127 | 0x040~0x07F | Base Configuration and DRAM Parameters |
| 2 | 128~191 | 0x080~0x0BF | ... |
| 3 | 192~239 | 0x0C0~0x0EF | Insert Annex A.0: Common |
| | 240~255 | 0x0D0~0x0FF | Insert Annex A.1: Solder Down |
| 4 | 256~319 | 0x100~0x13F | Insert Annex A.1: Solder Down |
| 5 | 320~383 | 0x140~0x17F | Insert Annex A.1: Solder Down |
| 6 | 384~447 | 0x180~0x1BF | Insert Annex A.1: Solder Down |
| 7~15 | 448~1023 | 0x180~0x3FF | ... |

## 4.2    UDIMM Overlay Schema

Key Byte 2 contains value 0x12 (DDR5)
Key Byte 3 contains any of the following values:

- 0x02, UDIMM
- 0x03, SODIMM

| Block | Range | | Description |
|---|---|---|---|
| 0 | 0~63 | 0x000~0x03F | Base Configuration and DRAM Parameters |
| 1 | 64~127 | 0x040~0x07F | Base Configuration and DRAM Parameters |
| 2 | 128~191 | 0x080~0x0BF | ... |
| 3 | 192~239 | 0x0C0~0x0EF | Insert Annex A.0: Common |
| | 240~255 | 0x0D0~0x0FF | Insert Annex A.2: Unbuffered Memory Module Types |
| 4 | 256~319 | 0x100~0x13F | Insert Annex A.2: Unbuffered Memory Module Types |
| 5 | 320~383 | 0x140~0x17F | Insert Annex A.2: Unbuffered Memory Module Types |
| 6 | 384~447 | 0x180~0x1BF | Insert Annex A.2: Unbuffered Memory Module Types |
| 7~15 | 448~1023 | 0x180~0x3FF | ... |

## 4.3   RDIMM and LRDIMM Overlay Schema

Key Byte 2 contains value 0x12 (DDR5)
Key Byte 3 contains any of the following values:

- 0x01, RDIMM
- 0x04, LRDIMM

| Block | Range | | Description |
|---|---|---|---|
| 0 | 0~63 | 0x000~0x03F | Base Configuration and DRAM Parameters |
| 1 | 64~127 | 0x040~0x07F | Base Configuration and DRAM Parameters |
| 2 | 128~191 | 0x080~0x0BF | ... |
| 3 | 192~239 | 0x0C0~0x0EF | Insert Annex A.0: Common |
| | 240~255 | 0x0D0~0x0FF | Insert Annex A.3: Registered and Load Reduced Memory Module Types |
| 4 | 256~319 | 0x100~0x13F | Insert Annex A.3: Registered and Load Reduced Memory Module Types |
| 5 | 320~383 | 0x140~0x17F | Insert Annex A.3: Registered and Load Reduced Memory Module Types |
| 6 | 384~447 | 0x180~0x1BF | Insert Annex A.3: Registered and Load Reduced Memory Module Types |
| 7~15 | 448~1023 | 0x180~0x3FF | ... |

## 4.4   Reserved Overlay Schema

## 4.5   DDIMM Overlay Schema

Key Byte 2 contains value 0x12 (DDR5)
Key Byte 3 contains any of the following values:

- 0x07, DDIMM

| Block | Range | | Description |
|---|---|---|---|
| 0 | 0~63 | 0x000~0x03F | Base Configuration and DRAM Parameters |
| 1 | 64~127 | 0x040~0x07F | Base Configuration and DRAM Parameters |
| 2 | 128~191 | 0x080~0x0BF | ... |
| 3 | 192~239 | 0x0C0~0x0EF | Insert Annex A.0: Common |
| | 240~255 | 0x0D0~0x0FF | Insert Annex A.5: Differential Memory Module Types |
| 4 | 256~319 | 0x100~0x13F | Insert Annex A.5: Differential Memory Module Types |
| 5 | 320~383 | 0x140~0x17F | Insert Annex A.5: Differential Memory Module Types |
| 6 | 384~447 | 0x180~0x1BF | Insert Annex A.5: Differential Memory Module Types |
| 7~15 | 448~1023 | 0x180~0x3FF | ... |

## 4.6 NVDIMM-N Overlay Schema

Key Byte 2 contains value 0x12 (DDR5)
Key Byte 3 contains any of the following values:

• 0x9X, NVDIMM-N Hybrid

where X refers to the base memory architecture

| Block | Range | | Description |
|---|---|---|---|
| 0 | 0~63 | 0x000~0x03F | Base Configuration and DRAM Parameters |
| 1 | 64~127 | 0x040~0x07F | Base Configuration and DRAM Parameters |
| 2 | 128~191 | 0x080~0x0BF | ... |
| 3 | 192~239 | 0x0C0~0x0EF | Insert Annex A.0: Common |
| | 240~255 | 0x0D0~0x0FF | Insert Annex A.6: NVDIMM-N Memory Module Types |
| 4 | 256~319 | 0x100~0x13F | Insert Annex A.6: NVDIMM-N Memory Module Types |
| 5 | 320~383 | 0x140~0x17F | Insert Annex A.6: NVDIMM-N Memory Module Types |
| 6 | 384~447 | 0x180~0x1BF | Insert Annex A.6: NVDIMM-N Memory Module Types |
| 7~15 | 448~1023 | 0x180~0x3FF | ... |

## 4.7 NVDIMM-P Overlay Schema

Key Byte 2 contains value 0x12 (DDR5)
Key Byte 3 contains any of the following values:

• 0xAX, NVDIMM-P Hybrid

where X refers to the base memory architecture

| Block | Range | | Description |
|---|---|---|---|
| 0 | 0~63 | 0x000~0x03F | Base Configuration and DRAM Parameters |
| 1 | 64~127 | 0x040~0x07F | Base Configuration and DRAM Parameters |
| 2 | 128~191 | 0x080~0x0BF | ... |
| 3 | 192~239 | 0x0C0~0x0EF | Insert Annex A.0: Common |
| | 240~255 | 0x0D0~0x0FF | Insert Annex A.7: NVDIMM-P Memory Module Types |
| 4 | 256~319 | 0x100~0x13F | Insert Annex A.7: NVDIMM-P Memory Module Types |
| 5 | 320~383 | 0x140~0x17F | Insert Annex A.7: NVDIMM-P Memory Module Types |
| 6 | 384~447 | 0x180~0x1BF | Insert Annex A.7: NVDIMM-P Memory Module Types |
| 7~15 | 448~1023 | 0x180~0x3FF | ... |

After programming the SPD contents, suppliers of JEDEC compliant modules must set the write protect bits for SPD device blocks 0 through 7. See JESD300-5 SPD5118 Device Specification for details on the Write Protect command protocol.

# 5    Parsing the SPD

| Some Relevant SPD Bytes for Parsing | |
|---|---|
| **SPD Byte(s)** | **Definition** |
| 1 | SPD revision for the base section of the SPD |
| 2 | DRAM interface type presented or emulated |
| 3 | Memory module interface type |
| 0~125 | Base configuration and DRAM parameters |
| 192 | SPD revision for the module specific annex |
| 192~239 | Common module parameters |
| 240~445 | Module specific parameters |
| 510~511 | CRC for bytes 0~509 |

The system BIOS will acquire information from the SPD in order to properly configure the system's memory controller. It is assumed the BIOS will parse the SPD data in the order listed below.

Step 1: Parse Byte 1 - Verify the SPD contents.

The first step is to verify the validity of the contents of the SPD. Calculate and verify that the 16-bit cyclic redundancy check (CRC) for bytes 0~509 matches the CRC stored in bytes 510~511.

Step 2: Parse Byte 2 - Verify the installed DRAM type is supported.

The first step in parsing the SPD is to verify that the DRAM type installed is supported by looking at DRAM device type byte 2. While it is usually not possible to physically plug in the wrong memory type, for example a DDR3 module size or key location should prevent insertion into a DDR5 system, there are cases where byte 2 is used to prevent accidental use of an incompatible memory type.

Step 3: Parse Byte 1 - Verify SPD compatibility. See Section 6 - SPD Revision Progression

The SPD revision byte 1 "encoding" nibble may be used to force legacy systems to reject newer modules. This would typically only occur if a critical error were found in SPD encoding that would require a "fix". In this case, as in the case of an unsupported DRAM type, system initialization must be halted immediately.

The SPD revision stored in Byte 1 applies to all information for the module, including base information and module specific information. Each SPD revision exactly defines how many bytes are valid in all other SPD blocks. The number of supported bytes (or bits) may increase from one SPD revision to another within each block as indicated by the "additions" nibble of the SPD revision. For example, an SPD revision 1.3 has more bytes or bits defined than SPD revision 1.2.

This progression of SPD contents is important for the BIOS to DIMM compatibility model. An older system may have a BIOS that only understands SPD revision 1.2 encoding, so if a module is installed that contains revision 1.3 information, the system can accurately interpret all of the historical revision 1.2 information retained in the module that is the subset of the revision 1.3 specification. Similarly, if a module with SPD revision 1.1 information is installed, that same BIOS can interpret all information that was current at the time that SPD 1.1 was defined. Therefore, BIOSes must maintain a knowledge of the active information for each historical SPD revision in order to support older modules.

## 5    Parsing the SPD (cont'd)

New in DDR5 SPD contents definition is a separate revision byte for sections of the contents. For example, the revision byte 1 covers only the base configuration information blocks, bytes 0 to 127. There is a separate revision byte for other sections such as the standard module parameters in bytes 192-447. The same scheme of contents and additions nibbles is used in each of these sections to allow forward and backward compatibility. Having separate revision levels in various sections of the SPD contents allows those sections to change independently.

Step 4: Parse Byte 3 - Determine module type and appropriate overlays.

Key byte 3 for module type determines the subsequent use of overlay information. Byte 3 defines the host to module interface style: unbuffered, registered, load reduced, differential, or hybrid.   Standard module types are defined in Annexes A.x.

Step 5: Parse Bytes 0~127 - Make base configuration settings to memory interface based on these bytes.

All module types are required to read and interpret this block of data to set up the DRAM type, maximum operating frequency, the number of row, column, bank bits, write recovery time, etc. While these bytes primarily describe the timing of the DRAMs, timing represents the capabilities of the module and it may be necessary to downgrade the timing based on other factors including layout or support components on the module, such as registers.

Step 6: Parse Byte 192 - Module-specific annex SPD revision.

This byte determines the encoding level and additions level of the module specific bytes of the SPD. Separating the base section of the SPD and module specific section prevents unnecessary churn; for example, if the UDIMM annex changes, the SPD for RDIMM/LRDIMM module need not change.

Step 7: Parse Bytes 193~447 - Configure standard module memory interface.

These bytes will be referenced and used by the system as needed based on the Standard module type that was determined after Byte 3 was parsed as indicated in Step 2. Bytes 192~239 are common byte definitions for all module types as defined in Annex A.0. Bytes 240~447 are coded differently for each standard module type as defined in Annex A.x.

## 6    SPD Revision Progression

SPD Contents sections have a revision code for that section of the specification. This allows each section to evolve over time independently as much as possible. For example a system may need to parse one revision level for the Base Configuration section (bytes 0~127) and a different revision level for the Standard Module Parameters section (bytes 192-447).

| Hypothetical SPD Revision Progression Showing Revision Relationships | | |
|---|---|---|
| Section | Bytes | Revision Byte |
| Base Configuration and SDRAM Parameters | 0~127 | 1 |
| Standard Module Parameters | 192~447 | 192 |

The Additions Level is never reduced even after an increment of the Encoding Level. For example, if the current SPD revision level were 1.2 and a change in Encoding Level were approved, the next revision level would be 2.2. If additions to revision 2.2 were approved, the next revision would be 2.3. Changes in the Encoding Level are extremely rare, however, since they can create incompatibilities with older systems.

The exceptions to the above rule are the SPD revision levels used during development prior to the Revision 1.0 release. Revisions 0.0 through 0.9 are used to indicate sequential pre-production SPD revision levels, however the first production release will be Revision 1.0.

## 6 SPD Revision Progression (cont'd)

A hypothetical progression is shown to illustrate the relationship between feature updates, typographical error corrections, specification revisions, and dates. In order to reduce the number of module requalifications, policy is to issue SPD releases once per year, typically in June (publication to the JEDEC public site typically takes an additional 3 months from committee approval).

| | | | | SPD Byte 1 | SPD Byte 192 | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Year | Description | Document | Release Date | Base | Solder Down | UDIMM | RDIMM/LRDIMM | DDIMM | NVDIMM-N | NVDIMM-P |
| 0 | Initial SPD document release | JESD400-5 | June 2021 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 0.1 | 0.1 |
| 0 | Typographical fixes | JESD400-5 | September 2021 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 0.1 | 0.1 |
| 0 | Typographical fixes | JESD400-5 | December 2021 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 0.1 | 0.1 |
| 1 | Additions to base, RDIMM<br>First release NVDIMM-P<br>Pre-production additions to NVDIMM-N | JESD400-5A | June 2022 | 1.1 | 1.0 | 1.0 | 1.1 | 1.0 | 0.2 | 1.0 |
| 1 | Typographical fixes | JESD400-5A | September 2022 | 1.1 | 1.0 | 1.0 | 1.1 | 1.0 | 0.2 | 1.0 |
| 1 | Typographical fixes | JESD400-5A | March 2023 | 1.1 | 1.0 | 1.0 | 1.1 | 1.0 | 0.2 | 1.0 |
| 2 | Additions to base, RDIMM, DDIMM<br>Additions to NVDIMM-P<br>First release NVDIMM-N | JESD400-5B | June 2023 | 1.2 | 1.0 | 1.0 | 1.2 | 1.1 | 1.0 | 1.1 |
| 3 | Additions to base | JESD400-5C | June 2024 | 1.3 | 1.0 | 1.0 | 1.2 | 1.1 | 1.0 | 1.1 |
| 3 | Typographical fixes | JESD400-5C | March 2025 | 1.3 | 1.0 | 1.0 | 1.2 | 1.1 | 1.0 | 1.1 |
| 4 | Additions to base, NVDIMM-N | JESD400-5D | June 2025 | 1.4 | 1.0 | 1.0 | 1.2 | 1.1 | 1.1 | 1.1 |

## 7 Address Map

The following is the SPD address map for all DDR5 modules. It describes where the individual lookup table entries will be held in the serial non-volatile memory. Consistent with the definition of DDR5 generation SPD devices which have 16 individual write protection blocks of 64 bytes in length each, the SPD contents are aligned with these blocks as follows:

| Block | Range | | Description |
|---|---|---|---|
| 0 | 0~63 | 0x000~0x03F | Base Configuration and DRAM Parameters |
| 1 | 64~127 | 0x040~0x07F | Base Configuration and DRAM Parameters |
| 2 | 128~191 | 0x080~0x0BF | Reserved for future use |
| 3 | 192~239 | 0x0C0~0x0EF | Common Module Parameters -- See Annex A.0 for details |
| | 240~255 | 0x0D0~0x0FF | Standard Module Parameters -- See annexes A.x for details |
| 4 | 256~319 | 0x100~0x13F | Standard Module Parameters -- See annexes A.x for details |
| 5 | 320~383 | 0x140~0x17F | Standard Module Parameters -- See annexes A.x for details |
| 6 | 384~447 | 0x180~0x1BF | Standard Module Parameters -- See annexes A.x for details |
| 7 | 448~509 | 0x1C0~0x1FD | Reserved for future use |
| | 510~511 | 0x1FE~0x1FF | CRC for bytes 0~509 |
| 8 | 512~575 | 0x200~0x23F | Manufacturing Information |
| 9 | 576~639 | 0x240~0x27F | Manufacturing Information |
| 10 | 640~703 | 0x280~0x2BF | End User Programmable |
| 11 | 704~767 | 0x2C0~0x2FF | End User Programmable |
| 12 | 768~831 | 0x300~0x33F | End User Programmable |
| 13 | 832~895 | 0x340~0x37F | End User Programmable |
| 14 | 896~959 | 0x380~0x3BF | End User Programmable |
| 15 | 960~1023 | 0x3C0~0x3FF | End User Programmable |

After programming the SPD contents, suppliers of JEDEC compliant modules must set the write protect bits for SPD device blocks 0 through 7. See the SPD5xx8 Device Specification for details on the block protect command protocol.

## 7 Address Map (cont'd)

### 7.1 Blocks 0~1: Base Configuration and DRAM Parameters

Table 1 details the location of each byte in this block.

**Table 1 — Blocks 0~1: Base Configuration and DRAM Parameters**

| Byte Number | | Function Described | Notes |
|---|---|---|---|
| 0 | 0x000 | Number of Bytes in SPD Device and Beta Level | |
| 1 | 0x001 | SPD Revision for Base Configuration Parameters | |
| 2 | 0x002 | Key Byte / Host Bus Command Protocol Type | |
| 3 | 0x003 | Key Byte / Module Type | |
| 4 | 0x004 | First SDRAM Density and Package | 1 |
| 5 | 0x005 | First SDRAM Addressing | 1 |
| 6 | 0x006 | First SDRAM I/O Width | 1 |
| 7 | 0x007 | First SDRAM Bank Groups and Banks Per Bank Group | 1 |
| 8 | 0x008 | Second SDRAM Density and Package | 1 |
| 9 | 0x009 | Second SDRAM Addressing | 1 |
| 10 | 0x00A | Second SDRAM I/O Width | 1 |
| 11 | 0x00B | Second SDRAM Bank Groups and Banks Per Bank Group | 1 |
| 12 | 0x00C | SDRAM BL32 and Post Package Repair | 1 |
| 13 | 0x00D | SDRAM Duty Cycle Adjuster and Partial Array Self Refresh | 1 |
| 14 | 0x00E | SDRAM Fault Handling | 1 |
| 15 | 0x00F | Reserved | |
| 16 | 0x010 | SDRAM Nominal Voltage, VDD | 1 |
| 17 | 0x011 | SDRAM Nominal Voltage, VDDQ | 1 |
| 18 | 0x012 | SDRAM Nominal Voltage, VPP | 1 |
| 19 | 0x013 | SDRAM Timing | 1 |
| 20 | 0x014 | SDRAM Minimum Cycle Time ($t_{CKAVG}$min), Least Significant Byte | 1 |
| 21 | 0x015 | SDRAM Minimum Cycle Time ($t_{CKAVG}$min), Most Significant Byte | 1 |
| 22 | 0x016 | SDRAM Maximum Cycle Time ($t_{CKAVG}$max), Least Significant Byte | 1 |
| 23 | 0x017 | SDRAM Maximum Cycle Time ($t_{CKAVG}$max), Most Significant Byte | 1 |
| 24 | 0x018 | CAS Latencies Supported, First Byte | 1 |
| 25 | 0x019 | CAS Latencies Supported, Second Byte | 1 |
| 26 | 0x01A | CAS Latencies Supported, Third Byte | 1 |
| 27 | 0x01B | CAS Latencies Supported, Fourth Byte | 1 |
| 28 | 0x01C | CAS Latencies Supported, Fifth Byte | 1 |
| 29 | 0x01D | Reserved | 1 |
| 30 | 0x01E | SDRAM Minimum CAS Latency Time ($t_{AA}$min), Least Significant Byte | 1 |
| 31 | 0x01F | SDRAM Minimum CAS Latency Time ($t_{AA}$min), Most Significant Byte | 1 |
| 32 | 0x020 | SDRAM Minimum RAS to CAS Delay Time ($t_{RCD}$min), Least Significant Byte | 1 |
| 33 | 0x021 | SDRAM Minimum RAS to CAS Delay Time ($t_{RCD}$min), Most Significant Byte | 1 |
| NOTE 1   From DDR5 SDRAM datasheet. | | | |

**Table 1 — Blocks 0~1: Base Configuration and DRAM Parameters (cont'd)**

| Byte Number | | Function Described | Notes |
|---|---|---|---|
| 34 | 0x022 | SDRAM Minimum Row Precharge Delay Time ($t_{RP}$min), Least Significant Byte | 1 |
| 35 | 0x023 | SDRAM Minimum Row Precharge Delay Time ($t_{RP}$min), Most Significant Byte | 1 |
| 36 | 0x024 | SDRAM Minimum Active to Precharge Delay Time ($t_{RAS}$min), Least Significant Byte | 1 |
| 37 | 0x025 | SDRAM Minimum Active to Precharge Delay Time ($t_{RAS}$min), Most Significant Byte | 1 |
| 38 | 0x026 | SDRAM Minimum Active to Active/Refresh Delay Time ($t_{RC}$min), Least Significant Byte | 1 |
| 39 | 0x027 | SDRAM Minimum Active to Active/Refresh Delay Time ($t_{RC}$min), Most Significant Byte | 1 |
| 40 | 0x28 | SDRAM Minimum Write Recovery Time ($t_{WR}$min), Least Significant Byte | 1 |
| 41 | 0x29 | SDRAM Minimum Write Recovery Time ($t_{WR}$min), Most Significant Byte | 1 |
| 42 | 0x2A | SDRAM Minimum Refresh Recovery Delay Time ($t_{RFC1}$min, $t_{RFC1}$_slr min), Least Significant Byte | 1 |
| 43 | 0x2B | SDRAM Minimum Refresh Recovery Delay Time ($t_{RFC1}$min, $t_{RFC1}$_slr min), Most Significant Byte | 1 |
| 44 | 0x02C | SDRAM Minimum Refresh Recovery Delay Time ($t_{RFC2}$min, $t_{RFC2}$_slr min), Least Significant Byte | 1 |
| 45 | 0x02D | SDRAM Minimum Refresh Recovery Delay Time ($t_{RFC2}$min, $t_{RFC2}$_slr min), Most Significant Byte | 1 |
| 46 | 0x02E | SDRAM Minimum Refresh Recovery Delay Time ($t_{RFCsb}$min, $t_{RFCsb}$_slr min), Least Significant Byte | 1 |
| 47 | 0x02F | SDRAM Minimum Refresh Recovery Delay Time ($t_{RFCsb}$_dlr min), Most Significant Byte | 1 |
| 48 | 0x030 | SDRAM Minimum Refresh Recovery Delay Time, 3DS Different Logical Rank ($t_{RFC1}$_dlr min), Least Significant Byte | 1 |
| 49 | 0x031 | SDRAM Minimum Refresh Recovery Delay Time, 3DS Different Logical Rank ($t_{RFC1}$_dlr min), Most Significant Byte | 1 |
| 50 | 0x032 | SDRAM Minimum Refresh Recovery Delay Time, 3DS Different Logical Rank ($t_{RFC2}$_dlr min), Least Significant Byte | 1 |
| 51 | 0x033 | SDRAM Minimum Refresh Recovery Delay Time, 3DS Different Logical Rank ($t_{RFC2}$_dlr min), Most Significant Byte | 1 |
| 52 | 0x034 | SDRAM Minimum Refresh Recovery Delay Time, 3DS Different Logical Rank ($t_{RFCsb}$_dlr min), Least Significant Byte | 1 |
| 53 | 0x035 | SDRAM Minimum Refresh Recovery Delay Time, 3DS Different Logical Rank ($t_{RFCsb}$_dlr min), Most Significant Byte | 1 |
| 54 | 0x036 | SDRAM Refresh Management, First Byte, First SDRAM | 1 |
| 55 | 0x037 | SDRAM Refresh Management, Second Byte, First SDRAM | 1 |
| 56 | 0x038 | SDRAM Refresh Management, First Byte, Second SDRAM | 1 |
| 57 | 0x039 | SDRAM Refresh Management, Second Byte, Second SDRAM | 1 |
| 58 | 0x03A | SDRAM Adaptive Refresh Management Level A, First Byte, First SDRAM | 1 |
| 59 | 0x03B | SDRAM Adaptive Refresh Management Level A, Second Byte, First SDRAM | 1 |
| 60 | 0x03C | SDRAM Adaptive Refresh Management Level A, First Byte, Second SDRAM | 1 |
| 61 | 0x03D | SDRAM Adaptive Refresh Management Level A, Second Byte, Second SDRAM | 1 |
| 62 | 0x03E | SDRAM Adaptive Refresh Management Level B, First Byte, First SDRAM | 1 |
| 63 | 0x03F | SDRAM Adaptive Refresh Management Level B, Second Byte, First SDRAM | 1 |
| 64 | 0x040 | SDRAM Adaptive Refresh Management Level B, First Byte, Second SDRAM | 1 |
| 65 | 0x041 | SDRAM Adaptive Refresh Management Level B, Second Byte, Second SDRAM | 1 |
| NOTE 1    From DDR5 SDRAM datasheet. | | | |

### Table 1 — Blocks 0~1: Base Configuration and DRAM Parameters (cont'd)

| Byte Number | | Function Described | Notes |
|---|---|---|---|
| 66 | 0x042 | SDRAM Adaptive Refresh Management Level C, First Byte, First SDRAM | 1 |
| 67 | 0x043 | SDRAM Adaptive Refresh Management Level C, Second Byte, First SDRAM | 1 |
| 68 | 0x044 | SDRAM Adaptive Refresh Management Level C, First Byte, Second SDRAM | 1 |
| 69 | 0x045 | SDRAM Adaptive Refresh Management Level C, Second Byte, Second SDRAM | 1 |
| 70 | 0x046 | SDRAM Minimum Active to Active Command Delay Time, Same Bank Group ($t_{RRD\_L}$min), Least Significant Byte | 1 |
| 71 | 0x047 | SDRAM Minimum Active to Active Command Delay Time, Same Bank Group ($t_{RRD\_L}$min), Most Significant Byte | 1 |
| 72 | 0x048 | SDRAM Minimum Active to Active Command Delay Time, Same Bank Group ($t_{RRD\_L}$min), Lower Clock Limit | 1 |
| 73 | 0x049 | SDRAM Minimum Read to Read Command Delay Time, Same Bank Group ($t_{CCD\_L}$min), Least Significant Byte | 1 |
| 74 | 0x04A | SDRAM Minimum Read to Read Command Delay Time, Same Bank Group ($t_{CCD\_L}$min), Most Significant Byte | 1 |
| 75 | 0x04B | SDRAM Minimum Read to Read Command Delay Time, Same Bank Group ($t_{CCD\_L}$min), Lower Clock Limit | 1 |
| 76 | 0x04C | SDRAM Minimum Write to Write Command Delay Time, Same Bank Group ($t_{CCD\_L\_WR}$min), Least Significant Byte | 1 |
| 77 | 0x04D | SDRAM Minimum Write to Write Command Delay Time, Same Bank Group ($t_{CCD\_L\_WR}$min), Most Significant Byte | 1 |
| 78 | 0x04E | SDRAM Minimum Write to Write Command Delay Time, Same Bank Group ($t_{CCD\_L\_WR}$min), Lower Clock Limit | 1 |
| 79 | 0x04F | SDRAM Minimum Write to Write Command Delay Time, Second Write not RMW, Same Bank Group ($t_{CCD\_L\_WR2}$min), Least Significant Byte | 1 |
| 80 | 0x050 | SDRAM Minimum Write to Write Command Delay Time, Second Write not RMW, Same Bank Group ($t_{CCD\_L\_WR2}$min), Most Significant Byte | 1 |
| 81 | 0x051 | SDRAM Minimum Write to Write Command Delay Time, Second Write not RMW, Same Bank Group ($t_{CCD\_L\_WR2}$min), Lower Clock Limit | 1 |
| 82 | 0x052 | SDRAM Minimum Four Activate Window ($t_{FAW}$min), Least Significant Byte | 1 |
| 83 | 0x053 | SDRAM Minimum Four Activate Window ($t_{FAW}$min), Most Significant Byte | 1 |
| 84 | 0x054 | SDRAM Minimum Four Activate Window ($t_{FAW}$min), Lower Clock Limit | 1 |
| 85 | 0x055 | SDRAM Minimum Write to Read Command Delay Time, Same Bank Group, ($t_{CCD\_L\_WTR}$min), Least Significant Byte | 1 |
| 86 | 0x056 | SDRAM Minimum Write to Read Command Delay Time, Same Bank Group, ($t_{CCD\_L\_WTR}$min), Most Significant Byte | 1 |
| 87 | 0x057 | SDRAM Minimum Write to Read Command Delay Time, Same Bank Group, ($t_{CCD\_L\_WTR}$min), Lower Clock Limit | 1 |
| 88 | 0x058 | SDRAM Minimum Write to Read Command Delay Time, Different Bank Group, ($t_{CCD\_S\_WTR}$min), Least Significant Byte | 1 |
| 89 | 0x059 | SDRAM Minimum Write to Read Command Delay Time, Different Bank Group, ($t_{CCD\_S\_WTR}$min), Most Significant Byte | 1 |
| 90 | 0x05A | SDRAM Minimum Write to Read Command Delay Time, Different Bank Group, ($t_{CCD\_S\_WTR}$min), Lower Clock Limit | 1 |
| 91 | 0x05B | SDRAM Minimum Read to Precharge Command Delay Time, ($t_{RTP}$min), Least Significant Byte | 1 |
| NOTE 1    From DDR5 SDRAM datasheet. | | | |

**Table 1 — Blocks 0~1: Base Configuration and DRAM Parameters (cont'd)**

| Byte Number | | Function Described | Notes |
|---|---|---|---|
| 92 | 0x05C | SDRAM Minimum Read to Precharge Command Delay Time, ($t_{RTP}$min), Most Significant Byte | 1 |
| 93 | 0x05D | SDRAM Minimum Read to Precharge Command Delay Time, ($t_{RTP}$min), Lower Clock Limit | 1 |
| 94~127 | 0x05E~0x07F | Reserved -- must be coded as 0x00 | |
| NOTE 1   From DDR5 SDRAM datasheet. | | | |

## 7.2    Block 2: Reserved for Future Use

## 7.3    Blocks 3~6: Module Specific Parameters

Bytes 192~447 (0x0C0~0x1BF) parameters in this block are specific to the module type as selected by the contents of SPD Key Byte 3 bits 3~0. Refer to the appropriate annex for detailed byte descriptions.

## 7.4    Block 7: Reserved for Future Use

Bytes 192~509 are reserved and must be programmed as 0.

**Byte 510 (0x1FE): Cyclical Redundancy Code (CRC) for SPD Bytes 0~509, Least Significant Byte**
**Byte 511 (0x1FF): Cyclical Redundancy Code (CRC) for SPD Bytes 0~509, Most Significant Byte**

This two-byte field contains the calculated CRC for bytes 0~509 (0x000~0x1FD) in the SPD. The following algorithm and data structures (shown in C) are to be followed in calculating and checking the code.

```c
int Crc16 (char *ptr, int count)
{
  int crc, i;

  crc = 0;
  while (--count >= 0) {
    crc = crc ^ (int)*ptr++ << 8;
    for (i = 0; i < 8; ++i)
      if (crc & 0x8000)
        crc = crc << 1 ^ 0x1021;
      else
        crc = crc << 1;
  }
  return (crc & 0xFFFF);
}
char spdBytes[] = { SPD_byte_0, SPD_byte_1, ..., SPD_byte_N-1 };
int data16;

data16 = Crc16 (spdBytes, sizeof(spdBytes));
SPD_byte_510 = (char) (data16 & 0xFF);
SPD_byte_511 = (char) (data16 >> 8);
```

## 7.5    Blocks 8~9: Manufacturing Information

Bytes 512~639 (0x200~0x27F). Table 2 details the location of each byte in this block.

**Table 2 — Blocks 8~9 Byte Location**

| Byte Number | | Function Described | Notes |
|---|---|---|---|
| 512 | 0x200 | Module Manufacturer's ID Code, First Byte | 1 |
| 513 | 0x201 | Module Manufacturer's ID Code, Second Byte | 1 |
| 514 | 0x202 | Module Manufacturing Location | 1 |
| 515~516 | 0x203~0x204 | Module Manufacturing Date | 1 |
| 517~520 | 0x205~0x208 | Module Serial Number | 1 |
| 521~550 | 0x209~0x226 | Module Part Number | 1 |
| 551 | 0x227 | Module Revision Code | |
| 552 | 0x228 | DRAM Manufacturer's ID Code, First Byte | 1 |
| 553 | 0x229 | DRAM Manufacturer's ID Code, Second Byte | 1 |
| 554 | 0x22A | DRAM Stepping | |
| 555~637 | 0x22B~0x27D | Module Manufacturer's Specific Data | |
| NOTE1    Required byte. | | | |

## 7.6    Blocks 10~15: End User Programmable

Bytes 640~1023 (0x280~0x3FF) Bytes in these blocks are reserved for use by end users. Module manufacturers should not assert write protect on blocks 10~15 unless requested to do so by customers.

## 7.7    ASCII Decode Matrix for SPDs

Table 3 is a subset of the full ASCII standard which is used for coding bytes in the Serial Presence Detect EEPROM that require ASCII characters:

**Table 3 — Subset of Full ASCII Standard**

| First Hex Digit in Pair | Second Hex Digit in Pair | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
| 2 | Blank Space | | | | | | | | ( | ) | | | | - Dash | . Period | |
| 3 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | | | | | | |
| 4 | | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O |
| 5 | P | Q | R | S | T | U | V | W | X | Y | Z | | | | | _ Underscore |
| 6 | | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o |
| 7 | p | q | r | s | t | u | v | w | x | y | z | | | | | |

**Examples:**
- 0x20 = Blank Space
- 0x34 = '4'
- 0x41 = 'A'

| SPD Bytes 521~550 | |
|---|---|
| **Manufacturer's PN** | **Coded in ASCII** |
| 13M32734BCD-260Y | 31334D33323733344243442D323630592020 |

# 8    Details of Each Byte

## 8.1    Blocks 0 and 1: General Configuration Section: Bytes 0~127 (0x000~0x07F)

This clause contains defined parameters that are common to all DDR5 module types with DDR5 SDRAM as the media type, and provides "key bytes" to allow overlay of module specific information. These bytes are defined when Key Byte 2 contains 18 (0x12), DDR5 SDRAM.

### 8.1.1    Byte 0 (0x000): Number of Bytes in SPD Device and Beta Level

The least significant nibble of this byte describes the total number of bytes used by the module manufacturer for the SPD data and any (optional) specific supplier information. The byte count includes the fields for all required and optional data. Bits 6~4 describe the total size of the serial memory used to hold the Serial Presence Detect data. Bits 7,3~0 define the Beta Level of the SPD encoding.

| Number of Bytes in SPD Device | | |
|---|---|---|
| Bit 7 | Bits 6~4 | Bits 3~0 |
| Beta Level 4 | SPD Bytes Total | Beta Level 3~0 |
| See bits 3~0 | 000: Undefined<br>001: 256<br>010: 512<br>011: 1024 (e.g., SPD5118)<br>100: 2048 (e.g., ESPD5216)<br>All others reserved | Bits 7, 3~0: Beta Level<br>Values 0 to 31 |
| The 5-bit Beta Level field (bit 7 and bits 3~0) is the incremental documentation release level to track changes between major external publications of the SPD Contents specification, typically done once per year. This value shall be reset to 00000 upon each external release and will be visible only to JEDEC members between releases. The Beta Level applies to all sections of the document, including DRAM and module annex sections. | | |

### 8.1.2    Byte 1 (0x001): SPD Revision for Base Configuration Parameters

This byte describes the compatibility level of the encoding of the bytes contained in the SPD EEPROM, and the current collection of valid defined bytes. Software should examine the upper nibble (Encoding Level) to determine if it can correctly interpret the contents of the module SPD. The lower nibble (Additions Level) can optionally be used to determine which additional bytes or attribute bits have been defined; however, since any undefined additional byte must be encoded as 0x00 or undefined attribute bit must be defined as 0, software can safely detect additional bytes and use safe defaults if a zero encoding is read for these bytes.

## 8.1.2 Byte 1 (0x001): SPD Revision for Base Configuration Parameters (cont'd)

| SPD Revision for Base Configuration Parameters | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Production Status** | **SPD Revision** | **Encoding Level** | | | | **Additions Level** | | | | **Hex** |
| | | **Bit 7** | **Bit 6** | **Bit 5** | **Bit 4** | **Bit 3** | **Bit 2** | **Bit 1** | **Bit 0** | |
| **Pre-production** | Revision 0.0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 00 |
| | Revision 0.1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 01 |
| | ... | . | . | . | . | . | . | . | . | . |
| | Revision 0.9 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 09 |
| **Production** | Revision 1.0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 10 |
| | Revision 1.1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 11 |
| | ... | . | . | . | . | . | . | . | . | ... |
| **Undefined** | Undefined | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | FF |

The Additions Level is never reduced even after an increment of the Encoding Level. For example, if the current SPD revision level were 1.2 and a change in Encoding Level were approved, the next revision level would be 2.2. If additions to revision 2.2 were approved, the next revision would be 2.3. Changes in the Encoding Level are extremely rare, however, since they can create incompatibilities with older systems.

The exceptions to the above rule are the SPD revision levels used during development prior to the Revision 1.0 release. Revisions 0.0 through 0.9 are used to indicate sequential pre-production SPD revision levels, however the first production release will be Revision 1.0.

Revisions for each module specific overlay are coded in the same format so that each module type can evolve over time independent of the base configuration parameters section.

The Beta Level (SPD byte 0, bits 7,3~0) may also be used to decode changes made since the last external publication of the SPD Contents.

### 8.1.3 Byte 2 (0x002): Key Byte / Host Bus Command Protocol Type

This byte is the key byte used by the system BIOS to determine how to interpret all other bytes in the SPD EEPROM. The BIOS must check this byte first to ensure that the EEPROM data is interpreted correctly. This command protocol is often based on the interface parameters for a specific memory device such as a DDR5 SDRAM, however may also be a general media interface protocol such as DDR5 NVDIMM-P. Protocol extensions such as the use of registers or data buffers must be documented in Byte 3, the Module Type byte, and such differences comprehended and accounted for by the system and memory controller. Some non-DRAM solutions called "Hybrid" use an extension of a DRAM bus protocol, such as NVDIMM-N, and use the code for the emulated DRAM protocol in this byte, such as DDR4 SDRAM.

**Table 4 — Key Byte / Host Bus Command Protocol Type**

| Key Byte / Host Bus Command Protocol Type | | |
|---|---|---|
| Line # | Hex | SDRAM / Module Type Corresponding to Key Byte |
| 0 | 00 | Reserved |
| 1 | 01 | Fast Page Mode |
| 2 | 02 | EDO |
| 3 | 03 | Pipelined Nibble |
| 4 | 04 | SDRAM |
| 5 | 05 | ROM |
| 6 | 06 | DDR SGRAM |
| 7 | 07 | DDR SDRAM |
| 8 | 08 | DDR2 SDRAM |
| 9 | 09 | DDR2 SDRAM FB-DIMM |
| 10 | 0A | DDR2 SDRAM FB-DIMM PROBE |
| 11 | 0B | DDR3 SDRAM |
| 12 | 0C | DDR4 SDRAM |
| 13 | 0D | Reserved |
| 14 | 0E | DDR4E SDRAM |
| 15 | 0F | LPDDR3 SDRAM |
| 16 | 10 | LPDDR4 SDRAM |
| 17 | 11 | LPDDR4X SDRAM |
| 18 | 12 | DDR5 SDRAM |
| 19 | 13 | LPDDR5 SDRAM |
| 20 | 14 | DDR5 NVDIMM-P |
| 21 | 15 | LPDDR5X SDRAM |
| - | - | - |
| 253 | FD | Reserved |
| 254 | FE | Reserved |
| 255 | FF | Reserved |

### 8.1.4    Byte 3 (0x003): Key Byte / Module Type

This byte is a Key Byte used to index the module specific section of the SPD from bytes 512~639. Byte 3 bits 3~0 identifies the SDRAM memory module type, and bits 7~4 describe hybrid memory extensions.

Some modules may have no base memory, but will have only a secondary memory type. For example, a Flash-only memory module. These are classified as "hybrid" for the purposes of interpreting the SPD. Where base memory parameters apply to this class of hybrid module, these will be documented with those bytes in the base section.

.

| Key Byte / Module Type | | |
|---|---|---|
| **Bit 7** | **Bits 6~4** | **Bits 3~0** |
| **Hybrid** | **Hybrid Media** | **Base Module Type** |
| 0: Not hybrid<br>(Module is DRAM only)<br>1: Hybrid module<br>(See bits 6~4 for hybrid type) | 000: Not hybrid<br>001: NVDIMM-N Hybrid<br>010: NVDIMM-P Hybrid<br>All other codes reserved | 0000: Reserved<br>0001: RDIMM<br>0010: UDIMM<br>0011: SODIMM<br>0100: LRDIMM<br>0101: Reserved<br>0110: Reserved<br>0111: Reserved<br>1000: Reserved<br>1001: Reserved<br>1010: DDIMM<br>1011: Solder down<br>1100: Reserved<br>1101: Reserved<br>1110: Reserved<br>1111: Reserved |
| Base Module Type Definitions:<br>Solder down: Direct attachment to memory controller<br>RDIMM:  Registered Dual In-Line Memory Module<br>UDIMM:  Unbuffered Dual In-Line Memory Module<br>SODIMM:  Unbuffered Small Outline Dual In-Line Memory Module<br>LRDIMM:  Load Reduced Dual In-Line Memory Module<br>DDIMM:  Differential Dual In-Line Memory Module<br><br>Hybrid Memory Type Definitions:<br>NVDIMM-N:  Non-Volatile Dual In-Line Memory Module, Hybrid module with a DDR5 SDRAM interface with one or more non-DRAM components for data storage<br>NVDIMM-P:  Non-Volatile Dual In-Line Memory Module, Hybrid module with a DDR5 NVDIMM-P interface with one or more non-DRAM components for data storage | | |

**Examples:**

0x01: RDIMM, no hybrid memory present

0x91: RDIMM, NVDIMM-N hybrid memory present

0x94: LRDIMM, NVDIMM-N hybrid memory present

0x0B = soldered down memory, no hybrid memory present

## 8.1.5    Byte 4 (0x004): First SDRAM Density and Package

This byte describes the SDRAM package type, loading as seen by the system, and device density in Gbits. These values comes from the DDR5 SDRAM data sheet, JESD79-5. This byte applies to the even rank SDRAMs for asymmetrical memory configurations, or for all SDRAMs for symmetrical configurations.

| First SDRAM Density and Package ||
| --- | --- |
| **Bits 7~5** | **Bits 4~0** |
| **Die Per Package** | **SDRAM Density Per Die** |
| 000:  1 die; Monolithic SDRAM<br>001:  Reserved<br>010: 2 die; 2H 3DS<br>011: 4 die; 4H 3DS<br>100: 8 die; 8H 3DS<br>101: 16 die; 16H 3DS<br><br>All others reserved | 00000: No memory; not defined<br>00001: 4 Gb<br>00010: 8 Gb<br>00011: 12 Gb<br>00100: 16 Gb<br>00101: 24 Gb<br>00110: 32 Gb<br>00111: 48 Gb<br>01000: 64 Gb<br><br>All others reserved |

## 8.1.6    Byte 5 (0x005): First SDRAM Addressing

This byte describes the number SDRAM column and row address bits. These values come from the DDR5 SDRAM data sheet, JESD79-5. This byte applies to the even rank SDRAMs for asymmetrical memory configurations, or for all SDRAMs for symmetrical configurations.

| First SDRAM Addressing ||
| --- | --- |
| **Bits 7~5** | **Bits 4~0** |
| **First SDRAM Column Address Bits** | **First SDRAM Row Address Bits** |
| 000: 10 columns<br>001: 11 columns<br><br>All others reserved | 00000: 16 rows<br>00001: 17 rows<br>00010: 18 rows<br><br>All others reserved |

### 8.1.7 Byte 6 (0x006): First SDRAM I/O Width

This byte describes the number SDRAM I/O bits (DQ) and the number of bank groups. These values come from the DDR5 SDRAM data sheet, JESD79-5. This byte applies to the even rank SDRAMs for asymmetrical memory configurations, or for all SDRAMs for symmetrical configurations.

| First SDRAM I/O Width | |
|---|---|
| **Bits 7~5** | **Bits 4~0** |
| **SDRAM I/O Width** | **Reserved** |
| 000: x4<br>001: x8<br>010: x16<br>011: x32<br><br>All others reserved | Reserved; must be coded as 00000 |

### 8.1.8 Byte 7 (0x007): First SDRAM Bank Groups and Banks Per Bank Group

This byte describes the number SDRAM banks per bank group. These values come from the DDR5 SDRAM data sheet, JESD79-5. This byte applies to the even rank SDRAMs for asymmetrical memory configurations, or for all SDRAMs for symmetrical configurations.

| First SDRAM Bank Groups and Banks Per Bank Group | | |
|---|---|---|
| **Bits 7~5** | **Bits 4~3** | **Bits 2~0** |
| **First SDRAM Bank Groups** | **Reserved** | **First SDRAM Banks Per Bank Group** |
| 000: 1 bank group<br>001: 2 bank groups<br>010: 4 bank groups<br>011: 8 bank groups<br><br>All others reserved | Reserved; must be coded as 00 | 000: 1 bank per bank group<br>001: 2 banks per bank group<br>010: 4 banks per bank group<br><br>All others reserved |

### 8.1.9    Byte 8 (0x008): Second SDRAM Density and Package

This byte describes the SDRAM package type, loading as seen by the system, and device density in Gbits. These values come from the DDR5 SDRAM data sheet, JESD79-5. This byte applies to the odd rank SDRAMs for asymmetrical memory configurations only.

| Second SDRAM Density and Package | |
|---|---|
| **Bits 7~5** | **Bits 4~0** |
| **Die Per Package** | **SDRAM Density Per Die** |
| 000:  1 die; Monolithic SDRAM<br>001:  Reserved<br>010: 2 die; 2H 3DS<br>011: 4 die; 4H 3DS<br>100: 8 die; 8H 3DS<br>101: 16 die; 16H 3DS<br><br>All others reserved | 00000: No memory; not defined<br>00001: 4 Gb<br>00010: 8 Gb<br>00011: 12 Gb<br>00100: 16 Gb<br>00101: 24 Gb<br>00110: 32 Gb<br>00111: 48 Gb<br>01000: 64 Gb<br><br>All others reserved |
| NOTE    This byte is defined only for asymmetrical memory systems; for symmetrical systems this byte must be coded as 0x00. | |

### 8.1.10    Byte 9 (0x009): Second SDRAM Addressing

This byte describes the number SDRAM column and row address bits. This byte describes the number SDRAM bank groups. These values come from the DDR5 SDRAM data sheet, JESD79-5. This byte applies to the odd rank SDRAMs for asymmetrical memory configurations only.

| Second SDRAM Addressing | |
|---|---|
| **Bits 7~5** | **Bits 4~0** |
| **Second SDRAM Column Address Bits** | **Second SDRAM Row Address Bits** |
| 000: 10 columns<br>001: 11 columns<br><br>All others reserved | 00000: 16 rows<br>00001: 17 rows<br>00010: 18 rows<br><br>All others reserved |
| NOTE    This byte is defined only for asymmetrical memory systems; for symmetrical systems this byte must be coded as 0x00. | |

### 8.1.11 Byte 10 (0x00A): Secondary SDRAM I/O Width

This byte describes the number SDRAM I/O bits (DQ) and the number of bank groups. These values come from the DDR5 SDRAM data sheet, JESD79-5. This byte applies to the odd rank SDRAMs for asymmetrical memory configurations only.

| Second SDRAM I/O Width | |
|---|---|
| **Bits 7~5** | **Bits 4~0** |
| **SDRAM I/O Width** | **Reserved** |
| 000: x4<br>001: x8<br>010: x16<br>011: x32<br><br>All others reserved | Reserved; must be coded as 00000 |
| NOTE    This byte is defined only for asymmetrical memory systems; for symmetrical systems this byte must be coded as 0x00. | |

### 8.1.12 Byte 11 (0x00B): Second SDRAM Bank Groups and Banks Per Bank Group

This byte describes the number SDRAM banks per bank group. These values come from the DDR5 SDRAM data sheet, JESD79-5. This byte applies to the odd rank SDRAMs for asymmetrical memory configurations only.

| Second SDRAM Bank Groups and Banks Per Bank Group | | |
|---|---|---|
| **Bits 7~5** | **Bits 4~3** | **Bits 2~0** |
| **Second SDRAM Bank Groups** | **Reserved** | **Second SDRAM Banks Per Bank Group** |
| 000: 1 bank group<br>001: 2 bank groups<br>010: 4 bank groups<br>011: 8 bank groups<br><br>All others reserved | Reserved; must be coded as 00 | 000: 1 bank per bank group<br>001: 2 banks per bank group<br>010: 4 banks per bank group<br><br>All others reserved |
| NOTE    This byte is defined only for asymmetrical memory systems; for symmetrical systems this byte must be coded as 0x00. | | |

### 8.1.13 Byte 12 (0x00C): SDRAM BL32 and Post Package Repair

This byte defines support for SDRAM features for burst length 32 (BL32) support and post package repair (PPR). These values come from the DDR5 SDRAM data sheet, JESD79-5.

| SDRAM BL32 and Post Package Repair Features | | | |
|---|---|---|---|
| **Bit 7** | **Bit 6** | **Bit 5** | **Bit 4** |
| **sPPR Granularity** | **Reserved** | **sPPR Undo/Lock** | **BL32** |
| 0 = One repair element per bank group<br>1 = One repair element per bank | Reserved; must be coded as 0 | 0: sPPR Undo/Lock not supported<br>1: sPPR Undo/Lock supported | 0: Burst length 32 not supported<br>1: Burst length 32 supported |
| **Bits 3~2** | | **Bit 1** | **Bit 0** |
| **Reserved** | | **MBIST/mPPR** | **Reserved** |
| Reserved; must be coded as 00 | | 0: MBIST/mPPR not supported<br>1: MBIST/mPPR supported | Reserved; must be coded as 0 |
| ACRONYMS:<br><br>MBIST = Memory Built-In Self Test<br>PPR = Post Package Repair<br>hPPR = Hard PPR<br>sPPR = Soft PPR<br>mPPR = MBIST PPR | | | |

### 8.1.14 Byte 13 (0x00D): SDRAM Duty Cycle Adjuster and Partial Array Self Refresh

This byte describes the options for the DDR5 SDRAM duty cycle adjuster (DCA).

| SDRAM Duty Cycle Adjuster | | | |
|---|---|---|---|
| **Bits 7~5** | **Bit 4** | **Bits 3~2** | **Bits 1~0** |
| **Reserved** | **PASR** | **Reserved** | **DCA Types Supported** |
| Reserved; must be coded as 0000 | 0: PASR not supported<br>1: PASR supported | Reserved; must be coded as 00 | 00 = Device does not support DCA<br>01 = Device supports DCA for single/2-phase internal clock(s)<br>10 = Device supports DCA for 4-phase internal clock(s)<br>11 = Reserved |
| ACRONYMS:<br><br>PASR = Partial Array Self Refresh<br>DCA = Duty Cycle Adjuster | | | |

### 8.1.15    Byte 14 (0x00E): SDRAM Fault Handling and Temperature Sense

This byte defines support for SDRAM fault handling features and for wide on-die temperature sensing range (see MR4). This value comes from the DDR5 SDRAM data sheet, JESD79-5.

| SDRAM Fault Handling Features and Temperature Sense | | | |
|---|---|---|---|
| Bits 7~4 | | | |
| Reserved; must be coded as 0000 | | | |
| Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Wide Temperature Sense | x4 RMW/ECS Writeback Suppression | x4 RMW/ECS Writeback Suppression MR Selector | Bounded Fault |
| 0: Wide temperature sense and reporting not supported<br>1: Wide temperature sense and reporting supported | 0: Writeback suppression not supported<br>1: Writeback suppression supported | 0: Writeback suppression control in MR9<br>1: Writeback suppression control in MR15 | 0: Bounded Fault not supported<br>1: Bounded Fault supported |
| ACRONYMS:<br><br>ECS = Error Check Scrub<br>RMW = Read-Modify-Write<br>MR = Mode Register | | | |

### 8.1.16    Byte 15 (0x00F): Reserved

Reserved - must be coded as 0x00.

### 8.1.17    Byte 16 (0x010): SDRAM Nominal Voltage, VDD

This byte describes the voltage levels for the SDRAM VDD supply only.

'Operable' is defined as the VDD voltage at which module operation is allowed using the performance values programmed in the SPD.

'Endurant' is defined as the VDD voltage at which the module may be powered without adversely affecting the life expectancy or reliability. Operation is not supported at this voltage.

| SDRAM Nominal Voltage, VDD | | |
|---|---|---|
| Nominal | Operable | Endurant |
| Bits 7~4 | Bits 3~2 | Bits 1~0 |
| 0000: 1.1 V<br>All others reserved | 00: 1.1 V<br>All others reserved | 00: 1.1 V<br>All others reserved |

### 8.1.18    Byte 17 (0x011): SDRAM Nominal Voltage, VDDQ

This byte describes the voltage levels for the SDRAM VDDQ supply only.

'Operable', 'Endurant': See Byte 16.

| SDRAM Nominal Voltage, VDDQ | | |
|---|---|---|
| Nominal | Operable | Endurant |
| Bits 7~4 | Bits 3~2 | Bits 1~0 |
| 0000: 1.1 V<br>All others reserved | 00: 1.1 V<br>All others reserved | 00: 1.1 V<br>All others reserved |

### 8.1.19    Byte 18 (0x012): SDRAM Nominal Voltage, VPP

This byte describes the voltage levels for the SDRAM VPP supply only.

'Operable', 'Endurant': See Byte 16.

| SDRAM Nominal Voltage, VPP | | |
|---|---|---|
| Nominal | Operable | Endurant |
| Bits 7~4 | Bits 3~2 | Bits 1~0 |
| 0000: 1.8 V<br>All others reserved | 00: 1.8 V<br>All others reserved | 00: 1.8 V<br>All others reserved |

### 8.1.20    Byte 19 (0x013): SDRAM Timing

This byte describes whether the SDRAM uses only standard core timings per JESD79-5, or non-standard timings for markets, such as overclocking systems. Core timings refer to tCKmin, tAAmin, tRCDmin, and tRPmin. This information assists in determining the most robust algorithm for calculating valid CAS latency settings.

| SDRAM Timing | |
|---|---|
| Reserved | Non-Standard Core Timing |
| Bits 7~1 | Bit 0 |
| Reserved; must be coded as 0000000 | 0 = SDRAM uses standard core timings per JESD79-5<br>1 = SDRAM supports non-standard core timings |

## 8.2    Timebases

Timebases in the DDR5 SPD are simplified compared to previous generations. Timing for parameters is expressed in either picosecond (ps) or nanosecond (ns) units. When programming memory controllers, it is often necessary to convert these timings to integer numbers of clocks (nCK) units. The algorithm for this conversion is described next.

## 8.3    Algorithms For Converting Time-based Parameters to Clock Counts

DDR5 SDRAM timing parameters are typically expressed in time units such as ns or ps, however digital logic design requires that these parameters be converted to a fixed number of clock periods "nCK" at the application frequency. The primary algorithm for this conversion is called the Rounding Algorithm, and it accounts for factors including errors caused by rounding and by digital integer expression or math calculation accuracy. The Rounding Algorithm is used for most parameters such as tRCDmin, tRCmin, tRFCmin, etc. A separate conversion called the CL Algorithm is defined specifically for determination of valid CAS Latency settings and accounts for factors including even latency requirements and unsupported CL settings.

The DDR5 SDRAM data sheet, JESD79-5, defines a baseline setting for timing that is common across vendors. However, specific devices in the industry may exceed the baseline timings for target markets such as point-to-point or overclocker systems. For these devices, the CL Algorithm may not apply, so specific rules are defined for determining valid CL settings. For non-standard SDRAMs, the memory supplier is expected to test for the additional boundary conditions created.

The guidelines for calculating CAS Latency are as follows:

- If Non-Standard Core Timing (SPD byte 19 bit 0) = 0 (i.e., the DRAM supports all JEDEC standard core timings), the preferred method for calculating valid CAS Latency settings is to use the CL Algorithm as it accounts for operation across a broad range of data rates including full speed or downbinning.
- Systems may prefer to use other methods to calculate CAS Latency settings, including applying the Rounding Algorithm on tAAmin (SPD bytes 30~31) to determine the desired CL setting, rounding up if necessary to ensure that only even nCK values are used, and applying the CAS Latency Mask (SPD bytes 24~28) to find a supported CL setting. If Non-Standard Core Timing (SPD byte 19 bit 0) = 1, the DRAM may support one or more exceptions or extensions of JEDEC standard core timings in which case other methods such as the Rounding Algorithm method described here are preferred to the CL Algorithm. Contact the memory supplier for details on supported CAS Latency conditions and recommended practices.

### 8.3.1    Rounding Algorithm

Software algorithms for calculation of timing parameters are subject to rounding errors from many sources. For example, a system may use a memory clock with a nominal frequency of 2200 MHz (4400 MT/s) for the DDR5-4400 speed bin, which mathematically yields ~~or~~ a clock period tCK(AVG) of 0.454545... ns. In most cases, it is impossible to express all digits after the decimal point exactly, and rounding must be used. The DDR5 SDRAM standard establishes a minimum granularity for timing parameters of 1 ps.

Rules for rounding must be defined to allow optimization of device performance without violating device parameters. All timing parameters specified in the time domain (ns, ps, etc.) which must then be converted to the clock domain (nCK units) shall be defined to align with these rules. The key point is, the minimum timing parameters shall generally use the same rounding rules used to define tCK(AVG)min. The resulting rounding algorithms rely on results that are within correction factors of device testing and specification to avoid losing performance due to rounding errors. These rules are:

## 8.3.1    Rounding Algorithm (cont'd)

- Minimum timing parameter values, including tCK(AVG)min, are to be rounded down and defined to 1 ps of accuracy in the DDR5 SDRAM specification based on the non-rounded nominal tCK(AVG)min for a given speed bin. If the nominal minimum timing parameter values require more than 1 ps of accuracy, they can be rounded down (faster) to the next 1 ps according to the rounding algorithms, and the DDR5 SDRAM is responsible for absorbing this small minimum parameter extension. In other words, the DDR5 SDRAM specification only lists the nominal minimum parameter values rounded down to the next 1ps. For example, this extends the DDR5-4400 tCK(AVG)min definition to be exactly 0.454 ns which is slightly smaller (faster) than the nominal memory clock period of 0.454545... ns by less than 1 ps.

- For minimum timing parameters, other than tCK(AVG)min, to avoid losing performance due to additional erroneous nCKs and to calculate the true real minimum values, their nominal values listed in the DDR5 SDRAM specification must be reduced (faster) by the same or greater % reduction (correction factor) that was used to define tCK(AVG)min. The DDR5 SDRAM is responsible for absorbing these parameter extensions. For example, tWRmin has a nominal value of 30.000ns, however, applying the 0.30% correction factor allows a more aggressive timing (for example, 29.910 ns) to be supported, which allows the intended smaller (faster) nCK value to be maintained when rounding tCK(AVG)min down to the next 1 ps. Parameter values defined as 0 do not need to be reduced by a correction factor, and therefore don't require these rounding algorithms.

- Round down only integer number math is commonly used in the industry to calculate nCK values. This second algorithm uses scaling by 1000 to allow use of integer math. Here, the nominal minimum parameter, in ps, is multiplied by the scaled correction factor (1000-3=997) prior to division by the application memory clock period, and 1 scaled by 1000 added to that result effectively rounds the result up. Division by 1000 undoes the scaling effects, resulting in a simple integer number of clocks as the final answer. The caveat is, effectively adding 1 prior to rounding down is mostly equivalent to rounding up except when the result is equal to an integer (whole number) in which case the result will not be rounded down as intended, and therefore performance would be lost. To address this, the largest correction factor of 0.28% needed for 3600 MHz (7200 MT/s) operation has been increased slightly to 0.30% in these rounding algorithms to account for this boundary condition. In the special case where a timing parameter is defined as 0, this rounding algorithm should not be used as it will result in lost performance.

$$nCK = truncate \left[ \frac{\left( \dfrac{truncate[\text{parameter\_nominal\_in\_ps}] \times 997}{truncate[\text{tCK(AVG)real\_in\_ps}]} \right) + 1000}{1000} \right]$$

**Example: Using round down only integer number math to convert $t_{WR(min)}$ from ps to nCK:**

```
// This algorithm reduces the nominal minimum timing parameter value by a 0.30% correction
// factor, adds 1 nCK, and rounds nCK down to the next integer value

int TwrMin, Correction, ClockPeriod, TempTwr, TempNck, TwrInNck;

TwrMin       = 30000;                      // tWRmin in ps
Correction   = 3                           // 0.30% per the rounding algorithm
ClockPeriod  = ApplicationTck;             // Clock period in ps is application specific
TempTwr      = TwrMin * (1000 - Correction);   // Apply correction factor, scaled by 1000
TempNck      = TempTwr / ClockPeriod ;     // Initial nCK calculation, scaled by 1000
TempNck      = TempNck + 1000;             // Add 1, scaled by 1000, to effectively round up
TwrInNck     = (int)(TempNck / 1000);
// Round down to next integer
```

## 8.3.1 Rounding Algorithm (cont'd)

**Table 5 — Example 2, Using Round Down Only Integer Number Math**

| DDR5 Device Operating at Standard Application Frequencies<br>Timing Parameter: *t*WR(min) = 30.000 ns = 30000 ps | | | | | |
|---|---|---|---|---|---|
| Application Speed Grade | Device *t*WR | Application tCK | Device tWR / Application tCK | Device tWR * (1000 - Correction) / Application tCK + 1000 | Truncate Corrected nCK / 1000 |
| | ps | ps | nCK (real) | scaled nCK (corrected) | nCK (integer) |
| 3200 | 30000 | 625 | 48.00 | 48856 | 48 |
| 3600 | 30000 | 555 | 54.05 | 54891 | 54 |
| 4000 | 30000 | 500 | 60.00 | 60820 | 60 |
| 4400 | 30000 | 454 | 66.08 | 66881 | 66 |
| 4800 | 30000 | 416 | 72.12 | 72899 | 72 |

## 8.3.2 CL Algorithm

The CL Algorithm specifically determines the valid CAS latency settings for SDRAMs with standard core timing values. This includes "downbinning", i.e., operating the SDRAMs at data rates below the maximum data rate supported by a specific device bin. This algorithm selects up to four valid CAS Latency values based on the combination of the tAAmin value for the installed SDRAM, the actual clock period tCKreal for the application data rate, and the valid CAS Latencies for standard JEDEC speeds. Some returned values may be RESERVED in which case a higher CAS Latency returned must be used – typically, the next higher CL value returned, though other values may be used to coordinate timing among various modules or memory channels. If no valid CL values are returned, i.e., all values are RESERVED, the algorithm has failed and other means must be used to determine a CAS Latency setting; contact your memory supplier for guidance. Similarly, if the four values returned are not sufficient for coordinating inter-module timings, other methods such as the CL Mask (SPD bytes 24~28) must be used to determine a common CL setting.

In pseudocode, the CL Algorithm is:

```
// Variables defined in the DDR5 SDRAM specification, JESD79-5
CorrFact = 0.30                                              // (%) Rounding Algorithm correction factor
ScaledCorrFact = 997                                         // Scaled correction factor (1000*(1-0.30%))
tCKreal =1011-952, 682-238                                   // (ps) Real application tCK(AVG) (1980-2100MT/s, 2933-8400MT/s)
tAAmin MONO=14000-17500, 3DS=16000-20000                    // (ps) From Speed Bin Tables and DIMM SPD bytes 30-31
tRCDtRPmin MONO=14000-17500, 3DS=14000-17500                // (ps) From Speed Bin Tables and DIMM SPD bytes 32-33 (tRCD=tRP)
tAAcorr = TRUNC(tAAmin*ScaledCorrFactor/1000)               // (ps) Corrected tAA(min) per the Rounding Algorithm rules
tRCDtRPcorr = TRUNC(tRCDtRPmin*ScaledCorrFactor/1000)       // (ps) Corrected tRCD(min), tRP(min) per the Rounding Algorithm
FUNC RA(targ) = TRUNC((targ*ScaledCorrFact/tCKstd+1000)/1000) // (nCK) Use Rounding Algorithm to convert bin target timing to nCK

// Round tCKreal down to the next faster standard frequency (tCK in ps)
IF (TRUNC(2000000/(2000*99%))>=TRUNC(tCKreal)>=TRUNC(2000000/(2000*105%)))  // Check for 1980-2100 nominal data rates
    tCKstd=TRUNC(2000000/2000)                                              // Assign standard 2000 tCK (ps)
ELSE IF (TRUNC(2000000/(2000+7*(133+(1/3))))>=TRUNC(tCKreal)>=TRUNC(2000000/3200))  // Check for 2933-3200 nominal data rates
    tCKstd=TRUNC(2000000/3200)                                              // Assign standard 3200 tCK (ps)
ELSE
    FOR (DataRateNom=3200; DataRateNom<=8000; DataRateNom=DataRateNom+400)  // Check for >3200-8000 nominal data rates
        IF (TRUNC(2000000/DataRateNom)>TRUNC(tCKreal)>=TRUNC(2000000/(DataRateNom+400)))
            tCKstd=TRUNC(2000000/(DataRateNom+400))                         // Assign standard 3600-8400 tCK (ps)
        ELSE
            tCKstd=RESERVED                                                 // No valid data rate found
```

## 8.3.2    CL Algorithm (cont'd)

```
// Timing targets (ps) that have been used to define the Speed Bin Tables
// MONO targets              // 3DS targets
BinAN_tAAtarg = 14000        BinAN_tAAtarg = 16000        // tAA target for AN bins
BinB__tAAtarg = 16000        BinB__tAAtarg = 18500        // tAA target for AN, B bins
BinBN_tAAtarg = 16000        BinBN_tAAtarg = 18500        // tAA target for AN, B, BN bins
BinC__tAAtarg = 17500        BinC__tAAtarg = 20000        // tAA target for AN, B, BN, C bins
BinAN_tRCDtRPtarg = 14000    BinAN_tRCDtRPtarg = 14000    // tRCD, tRP target for AN bins
BinBN_tRCDtRPtarg = 16000    BinBN_tRCDtRPtarg = 16000    // tRCD, tRP target for AN, B, BN bins
BinC__tRCDtRPtarg = 17500    BinC__tRCDtRPtarg = 17500    // tRCD, tRP target for AN, B, BN, C bins
IF (TRUNC(2000000/3600)>tCKstd)                          // tRCD, tRP target for B bins is frequency dependent
    BinB__tRCDtRPtarg = 16000 BinB__tRCDtRPtarg = 16000  // tRCD, tRP target for AN, B bins data rates faster than 3600
ELSE
    // 16250=(2000000/3200)*EVEN(TRUNC((BinB__tRCDtRPtarg*ScaledCorrFact/(2000000/3200)+1000)/1000))
    BinB__tRCDtRPtarg = 16250 BinB__tRCDtRPtarg = 16250  // tRCD, tRP target for AN, B bins for data rates 3600 and slower


// CL Algorithm using variables defined above
// Up to four valid CL's can be returned for a specific freq: CL(AN), CL(B), CL(BN), CL(C), depending on tAAmin, tRCDmin, tRPmin
// The B and BN bins return the same CL
// Only even CL's (not odd CL's) are valid per the DDR5 SDRAM specification
// nRCD, nRP are only even at standard native frequencies for the AN, BN bins (can be even or odd at intermediate frequencies)
// nRCD, nRP may be even or odd at standard native frequencies for the B, C bins (can be even or odd at intermediate frequencies)
IF (TRUNC(2000000/2000)=tCKstd)                          // CL22 is the only valid CL for 1980-2100 data rates
    CL(AN)=22                                            // Valid even CL for AN bins
    CL(B )=22                                            // Valid even CL for AN, B, bins
    CL(BN)=22                                            // Valid even CL for AN, B, BN bins
    CL(C )=22                                            // Valid even CL for AN, B, BN, C bins
ELSE IF (TRUNC(2000000/3200)>=tCKstd>=TRUNC(2000000/8400))     // Valid CL for 2933-8400 data rates
    IF ((EVEN(RA(BinAN_tAAtarg))*tCKstd>=tAAcorr)AND(EVEN(RA(BinAN_tRCDtRPtarg))*tCKstd>=tRCDtRPcorr)) // nRCD, nRP even only
        CL(AN)=EVEN(RA(BinAN_tAAtarg))                  // Valid even CL for AN bins
        CL(B )=EVEN(RA(BinB__tAAtarg))                  // Valid even CL for AN, B bins
        CL(BN)=EVEN(RA(BinBN_tAAtarg))                  // Valid even CL for AN, B, BN bins
        CL(C )=EVEN(RA(BinC__tAAtarg))                  // Valid even CL for AN, B, BN, C bins
    ELSE IF ((EVEN(RA(BinB__tAAtarg))*tCKstd>=tAAcorr)AND( (RA(BinB__tRCDtRPtarg))*tCKstd>=tRCDtRPcorr)) // nRCD, nRP even, odd
        CL(AN)=RESERVED                                 // Valid even CL for AN bins
        CL(B )=EVEN(RA(BinB__tAAtarg))                  // Valid even CL for AN, B bins
        CL(BN)=EVEN(RA(BinBN_tAAtarg))                  // Valid even CL for AN, B, BN bins
        CL(C )=EVEN(RA(BinC__tAAtarg))                  // Valid even CL for AN, B, BN, C bins
    ELSE IF ((EVEN(RA(BinBN_tAAtarg))*tCKstd>=tAAcorr)AND(EVEN(RA(BinBN_tRCDtRPtarg))*tCKstd>=tRCDtRPcorr)) // nRCD, nRP even only
        CL(AN)=RESERVED                                 // Valid even CL for AN bins
        CL(B )=RESERVED                                 // Valid even CL for AN, B bins
        CL(BN)=EVEN(RA(BinBN_tAAtarg))                  // Valid even CL for AN, B, BN bins
        CL(C )=EVEN(RA(BinC__tAAtarg))                  // Valid even CL for AN, B, BN, C bins
    ELSE IF ((EVEN(RA(BinC__tAAtarg))*tCKstd>=tAAcorr)AND( (RA(BinC__tRCDtRPtarg))*tCKstd>=tRCDtRPcorr)) // nRCD, nRP even, odd
        CL(AN)=RESERVED                                 // Valid even CL for AN bins
        CL(B )=RESERVED                                 // Valid even CL for AN, B bins
        CL(BN)=RESERVED                                 // Valid even CL for AN, B, BN bins
        CL(C )=EVEN(RA(BinC__tAAtarg))                  // Valid even CL for AN, B, BN, C bins
    ELSE                                                // No valid CL found (tAAmin, tRCDmin, tRPmin are too slow)
        CL(AN)=RESERVED                                 // Valid even CL for AN bins
        CL(B )=RESERVED                                 // Valid even CL for AN, B bins
        CL(BN)=RESERVED                                 // Valid even CL for AN, B, BN bins
        CL(C )=RESERVED                                 // Valid even CL for AN, B, BN, C bins
ELSE                                                    // No valid data rate found
    CL(AN)=RESERVED                                     // Valid even CL for AN bins
    CL(B )=RESERVED                                     // Valid even CL for AN, B bins
    CL(BN)=RESERVED                                     // Valid even CL for AN, B, BN bins

    CL(C )=RESERVED                                     // Valid even CL for AN, B, BN, C bins
```

## 8.4     SDRAM Minimum Cycle Time (t$_{CKAVG}$min)

### 8.4.1     Byte 20 (0x014): Least Significant Byte
###         Byte 21 (0x015): Most Significant Byte

This 16-bit word defines the minimum cycle time for the SDRAM module, in picoseconds (ps). This number applies to all applicable components on the module. This byte applies to SDRAM and support components as well as the overall capability of the DIMM. This value comes from the DDR5 SDRAM data sheet, JESD79-5, and supports component data sheets.

| SDRAM Minimum Cycle Time (t$_{CKAVG}$min) ps | |
|---|---|
| Byte 21 | Byte 20 |
| Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 65535 ps<br>0x0000: Reserved | |

## 8.5     SDRAM Maximum Cycle Time (t$_{CKAVG}$max)

### 8.5.1     Byte 22 (0x016): Least Significant Byte
###         Byte 23 (0x017): Most Significant Byte

This 16-bit word defines the minimum cycle time for the SDRAM module, in picoseconds (ps). This number applies to all applicable components on the module. This byte applies to SDRAM and support components as well as the overall capability of the DIMM. This value comes from the DDR5 SDRAM data sheet, JESD79-5, and supports component data sheets.

| SDRAM Maximum Cycle Time (t$_{CKAVG}$max) ps | |
|---|---|
| Byte 23 | Byte 22 |
| Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 65535 ps.<br>0x0000: Reserved | |

### 8.6    SDRAM CAS Latencies Supported

### 8.6.1    Byte 24 (0x018): First Byte
Byte 25 (0x019): Second Byte
Byte 26 (0x01A): Third Byte
Byte 27 (0x01B): Fourth Byte
Byte 28 (0x01C): Fifth Byte

These bytes define which CAS Latency (CL) values are supported with one bit per possible CAS Latency. A 1 in a bit position means that CL is supported, a 0 in that bit position means it is not supported. These values come from the DDR5 SDRAM, JESD79-5. CAS Latency Mask covers both DBI and non-DBI modes of operation.

| SDRAM CAS Latencies Supported | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Byte 24** | **Bit 7** | **Bit 6** | **Bit 5** | **Bit 4** | **Bit 3** | **Bit 2** | **Bit 1** | **Bit 0** |
| CL = | 34 | 32 | 30 | 28 | 26 | 24 | 22 | 20 |
| **Byte 25** | **Bit 7** | **Bit 6** | **Bit 5** | **Bit 4** | **Bit 3** | **Bit 2** | **Bit 1** | **Bit 0** |
| CL = | 50 | 48 | 46 | 44 | 42 | 40 | 38 | 36 |
| **Byte 26** | **Bit 7** | **Bit 6** | **Bit 5** | **Bit 4** | **Bit 3** | **Bit 2** | **Bit 1** | **Bit 0** |
| CL = | 66 | 64 | 62 | 60 | 58 | 56 | 54 | 52 |
| **Byte 27** | **Bit 7** | **Bit 6** | **Bit 5** | **Bit 4** | **Bit 3** | **Bit 2** | **Bit 1** | **Bit 0** |
| CL = | 82 | 80 | 78 | 76 | 74 | 72 | 70 | 68 |
| **Byte 28** | **Bit 7** | **Bit 6** | **Bit 5** | **Bit 4** | **Bit 3** | **Bit 2** | **Bit 1** | **Bit 0** |
| CL = | 98 | 96 | 94 | 92 | 90 | 88 | 86 | 84 |
| For each bit position, 0 means this CAS Latency is not supported, 1 means this CAS Latency is supported. | | | | | | | | |

**Example 1: DDR5-4400A SDRAM**
CAS latencies supported: 22, 26, 28, 30, 32, 36, 40

| SPD Byte | Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Hex Code |
|---|---|---|---|---|---|---|---|---|---|---|
| Byte 24 | CAS Latencies | 34 | 32 | 30 | 28 | 26 | 24 | 22 | 20 | |
| | CL Mask | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0x7A |
| Byte 25 | CAS Latencies | 50 | 48 | 46 | 44 | 42 | 40 | 38 | 36 | |
| | CL Mask | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0x05 |
| Byte 26 | CAS Latencies | 66 | 64 | 62 | 60 | 58 | 56 | 54 | 52 | |
| | CL Mask | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0x00 |
| Byte 27 | CAS Latencies | 82 | 80 | 78 | 76 | 74 | 72 | 70 | 68 | |
| | CL Mask | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0x00 |
| Byte 28 | CAS Latencies | 98 | 96 | 94 | 92 | 90 | 88 | 86 | 84 | |
| | CL Mask | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0x00 |

### 8.6.2    Byte 29 (0x01D)

Reserved - must be coded as 0x00.

## 8.7 SDRAM Minimum CAS Latency Time ($t_{AA}$min)

### 8.7.1 Byte 30 (0x01E): Least Significant Byte
### Byte 31 (0x01F): Most Significant Byte

This 16-bit word defines the minimum CAS Latency in picoseconds (ps). This value comes from the DDR5 SDRAM data sheet, JESD79-5.

| SDRAM Minimum CAS Latency Time ($t_{AA}$min) ps | |
|---|---|
| Byte 31 | Byte 30 |
| Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 65535 ps 0x0000: Reserved | |

## 8.8 SDRAM Minimum RAS to CAS Delay Time ($t_{RCD}$min)

### 8.8.1 Byte 32 (0x020): Least Significant Byte
### Byte 33 (0x021): Most Significant Byte

This 16-bit word defines the minimum SDRAM RAS to CAS Delay Time in picoseconds (ps). This value comes from the DDR5 SDRAM data sheet, JESD79-5.

| SDRAM Minimum RAS to CAS Delay Time ($t_{RCD}$min) ps | |
|---|---|
| Byte 33 | Byte 32 |
| Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 65535 ps. 0x0000: Reserved | |

## 8.9 SDRAM Minimum Row Precharge Delay Time ($t_{RP}$min)

### 8.9.1 Byte 34 (0x022): Least Significant Byte
### Byte 35 (0x023): Most Significant Byte

This 16-bit word defines the minimum SDRAM Row Precharge Delay Time in picoseconds (ps). This value comes from the DDR5 SDRAM data sheet, JESD79-5.

| SDRAM Minimum Row Precharge Delay Time ($t_{RP}$min) ps | |
|---|---|
| Byte 35 | Byte 34 |
| Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 65535 ps. 0x0000: Reserved | |

## 8.10 SDRAM Minimum Active to Precharge Delay Time (t$_{RAS}$min)

### 8.10.1 Byte 36 (0x024): Least Significant Byte
Byte 37 (0x025): Most Significant Byte

This 16-bit word defines the minimum SDRAM Active to Precharge Delay Time in picoseconds (ps). This value comes from the DDR5 SDRAM data sheet, JESD79-5.

| SDRAM Minimum Active to Precharge Delay Time (t$_{RAS}$min) ps | |
|---|---|
| Byte 37 | Byte 36 |
| Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 65535 ps<br>0x0000: Reserved | |

## 8.11 SDRAM Minimum Active to Active/Refresh Delay Time (t$_{RC}$min)

### 8.11.1 Byte 38 (0x026): Least Significant Byte
Byte 39 (0x027): Most Significant Byte

This 16-bit word defines the minimum SDRAM Active to Active/Refresh Delay Time in picoseconds (ps). This value comes from the DDR5 SDRAM data sheet, JESD79-5.

| SDRAM Minimum Active to Active/Refresh Delay Time (t$_{RC}$min) ps | |
|---|---|
| Byte 39 | Byte 38 |
| Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 65535 ps.<br>0x0000: Reserved | |

By DDR5 device design, t$_{RC}$min must always be greater than or equal to t$_{RAS}$min + t$_{RP}$min. When converting time in ns to nCK units using the appropriate rounding algorithms for these three parameters, systems must account for the overall design requirement and choose the larger of t$_{RC}$min or (t$_{RAS}$min + t$_{RP}$min) as a minimum setting.

## 8.12    SDRAM Minimum Write Recovery Time ($t_{WR}$min)

### 8.12.1    Byte 40 (0x028): Least Significant Byte
### Byte 41 (0x029): Most Significant Byte

This 16-bit word defines the minimum SDRAM Write Recovery Time in picoseconds (ps). This value comes from the DDR5 SDRAM data sheet, JESD79-5.

| SDRAM Minimum Write Recovery Time ($t_{WR}$min) ps | |
|---|---|
| Byte 41 | Byte 40 |
| Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 65535 ps<br>0x0000: Reserved | |

## 8.13    SDRAM Minimum Refresh Recovery Delay Time ($t_{RFC1}$min, $t_{RFC1\_}$slr min)

### 8.13.1    Byte 42 (0x02A): Least Significant Byte
### Byte 43 (0x02B): Most Significant Byte

This 16-bit word defines the minimum SDRAM Refresh Recovery Time Delay in nanoseconds (ns). $t_{RFC1}$ relates to monolithic SDRAMs, $t_{RFC1\_}$slr to 3DS SDRAMs. These values come from the DDR5 SDRAM data sheet, JESD79-5.

| SDRAM Minimum Refresh Recovery Delay Time ($t_{RFC1}$min, $t_{RFC1\_}$slr min) ns | |
|---|---|
| Byte 43 | Byte 42 |
| Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 65535 ns<br>0x0000 = Reserved | |
| NOTE    $t_{RFC1}$min relates to monolithic SDRAMs, $t_{RFC1\_}$slr min relates to 3DS SDRAMs. | |

## 8.14    SDRAM Minimum Refresh Recovery Delay Time ($t_{RFC2}$min, $t_{RFC2\_}$slr min)

### 8.14.1    Byte 44 (0x02C): Least Significant Byte
### Byte 45 (0x02D): Most Significant Byte

This 16-bit word defines the minimum SDRAM Refresh Recovery Time Delay in nanoseconds (ns). $t_{RFC2}$ relates to monolithic SDRAMs, $t_{RFC2\_}$slr to 3DS SDRAMs. These values come from the DDR5 SDRAM data sheet, JESD79-5.

| SDRAM Minimum Refresh Recovery Delay Time ($t_{RFC2}$min, $t_{RFC2\_}$slr min) ns | |
| --- | --- |
| **Byte 45** | **Byte 44** |
| **Bits 15~8** | **Bits 7~0** |
| Values defined from 1 to 65535 ns<br>0x0000 = Reserved | |
| NOTE    $t_{RFC2}$min relates to monolithic SDRAMs, $t_{RFC2\_}$slr min relates to 3DS SDRAMs. | |

## 8.15    SDRAM Minimum Refresh Recovery Delay Time ($t_{RFCsb}$min, $t_{RFCsb\_}$slr min)

### 8.15.1    Byte 46 (0x02E): Least Significant Byte
### Byte 47 (0x02F): Most Significant Byte

This 16-bit word defines the minimum SDRAM Refresh Recovery Time Delay in nanoseconds (ns). $t_{RFCsb}$ relates to monolithic SDRAMs, $t_{RFCsb\_}$slr to 3DS SDRAMs. These values come from the DDR5 SDRAM data sheet, JESD79-5.

| SDRAM Minimum Refresh Recovery Delay Time ($t_{RFCsb}$min) ns | |
| --- | --- |
| **Byte 47** | **Byte 46** |
| **Bits 15~8** | **Bits 7~0** |
| Values defined from 1 to 65535 ns<br>0x0000: Reserved | |
| NOTE    $t_{RFCsb}$min relates to monolithic SDRAMs, $t_{RFCsb\_}$slr min relates to 3DS SDRAMs. | |

### 8.16 SDRAM Minimum Refresh Recovery Delay Time, 3DS Different Logical Rank ($t_{RFC1\_}$dlr min)

#### 8.16.1 Byte 48 (0x030): Least Significant Byte
Byte 49 (0x031): Most Significant Byte

This 16-bit word defines the minimum SDRAM Refresh Recovery Time Delay in nanoseconds (ns). $t_{RFC1\_}$dlr relates to 3DS SDRAMs only. For monolithic SDRAMs, these bytes must be encoded as 0x0000. These values come from the DDR5 SDRAM data sheet, JESD79-5.

| SDRAM Minimum Refresh Recovery Delay Time ($t_{RFC1\_}$dlr min) ns | |
|---|---|
| Byte 49 | Byte 48 |
| Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 65535 ns<br>0x0000 = Reserved | |
| NOTE    These bytes are defined for 3DS only; for monolithic SDRAMs, code as 0x0000. | |

### 8.17 SDRAM Minimum Refresh Recovery Delay Time, 3DS Different Logical Rank ($t_{RFC2\_}$dlr min)

#### 8.17.1 Byte 50 (0x032): Least Significant Byte
Byte 51 (0x033): Most Significant Byte

This 16-bit word defines the minimum SDRAM Refresh Recovery Time Delay in nanoseconds (ns). $t_{RFC2\_}$dlr relates to 3DS SDRAMs only. For monolithic SDRAMs, these bytes must be encoded as 0x0000. These values come from the DDR5 SDRAM data sheet, JESD79-5.

| SDRAM Minimum Refresh Recovery Delay Time ($t_{RFC2\_}$dlr min) ns | |
|---|---|
| Byte 51 | Byte 50 |
| Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 65535 ns<br>0x0000 = Reserved | |
| NOTE    These bytes are defined for 3DS only; for monolithic SDRAMs, code as 0x0000. | |

### 8.18 SDRAM Minimum Refresh Recovery Delay Time, 3DS Different Logical Rank (t$_{RFCsb\_}$dlr min)

#### 8.18.1 Byte 52 (0x034): Least Significant Byte
Byte 53 (0x035): Most Significant Byte

This 16-bit word defines the minimum SDRAM Refresh Recovery Time Delay in nanoseconds (ns). t$_{RFCsb\_}$dlr relates to 3DS SDRAMs only. For monolithic SDRAMs, these bytes must be encoded as 0x0000. These values come from the DDR5 SDRAM data sheet, JESD79-5.

| SDRAM Minimum Refresh Recovery Delay Time (t$_{RFCsb\_}$dlr min) ns | |
|---|---|
| Byte 53 | Byte 52 |
| Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 65535 ns<br>0x0000 = Reserved | |
| NOTE  These bytes are defined for 3DS only; for monolithic SDRAMs, code as 0x0000 | |

## 8.19    Byte 54 (0x036): SDRAM Refresh Management, First Byte, First SDRAM
## Byte 55 (0x037): SDRAM Refresh Management, Second Byte, First SDRAM

These bytes represent the refresh management (RFM) settings for the SDRAMs used on this module. These are identical to the contents of the DDR5 SDRAM mode registers MR58 and MR59. These values come from the DDR5 SDRAM data sheet, JESD79-5. If all devices in a group (i.e., First or Second) are not matched, the value coded must represent the worst case for any device in the group.

| SDRAM Refresh Management (RFM), First Byte<br>Byte 54 (First SDRAM) | | | | |
|---|---|---|---|---|
| **Bits 7~5**<br>**(MR58:OP[7:5])** | | **Bits 4~1**<br>**(MR58:OP[4:1])** | | **Bit 0**<br>**(MR58:OP[0])** |
| **RAAMMT** | | **RAAIMT** | | **RFM Required** |
| **Normal** | **FGR** | **Normal** | **FGR** | |
| 000~010: Reserved<br>011: 3X<br>100: 4X<br>101: 5X<br>110: 6X<br>111: Reserved | 000~010: Reserved<br>011: 6X<br>100: 8X<br>101: 10X<br>110: 12X<br>111: Reserved | 0000-0011: Reserved<br>0100: 32<br>0101: 40<br>...<br>1001: 72<br>1010: 80<br>All other codings reserved | 0000-0011: Reserved<br>0100: 16<br>0101: 20<br>...<br>1001: 36<br>1010: 40<br>All other codings reserved | 0: Refresh management not required<br>1: Refresh management required |

NOTES:
RFM = Refresh Management
RAAMMT = Rolling Accumulated ACT Maximum Management Threshold
RAAIMT = Rolling Accumulated ACT Initial Management Threshold
Normal = Normal refresh mode
FGR = Fine granularity refresh mode
First SDRAM applies to all package ranks of SDRAMs in symmetrical configurations or even ranks in asymmetrical configurations
Second SDRAM applies to odd ranks in asymmetrical configurations only

| SDRAM Refresh Management (RFM), Second Byte<br>Byte 55 (First SDRAM) | | |
|---|---|---|
| **Bits 7~6**<br>**(MR59:OP[7:6])** | **Bits 5~4**<br>**(MR59:OP[5:4])** | **Bits 3~0**<br>**(MR59:OP[3:0])** |
| **RFM RAA Counter Decrement per REF command** | **ARFM Level** | **Reserved** |
| 00: RAAIMT<br>01: RAAIMT ÷ 2<br>Other codes reserved | Reserved; must be coded in the SPD as 00; modified by the host controller in the SDRAM to select ARFM Level | Reserved; must be coded as 0000 |

NOTES:
ARFM = Adaptive Refresh Management
RAA = Rolling Accumulated ACT
RAAIMT = Rolling Accumulated ACT Initial Management Threshold
First SDRAM applies to all package ranks of SDRAMs in symmetrical configurations or even ranks in asymmetrical configurations
Second SDRAM applies to odd ranks in asymmetrical configurations only
Selection of ARFM level is managed by writing to SDRAM mode register bits MR59:OP[5:4]

### 8.20 Byte 56 (0x038): SDRAM Refresh Management, First Byte, Second SDRAM
### Byte 57 (0x039): SDRAM Refresh Management, Second Byte, Second SDRAM

These bytes represent the refresh management (RFM) settings for the Second SDRAMs used on this module for odd ranks in asymmetrical assemblies.

| SDRAM Refresh Management (RFM), First Byte<br>Byte 56 (Second SDRAM) | | |
|---|---|---|
| Bits 7~5<br>(MR58:OP[7:5]) | Bits 4~1<br>(MR58:OP[4:1]) | Bit 0<br>(MR58:OP[0]) |
| RAAMMT | RAAIMT | RFM Required |
| See byte 54 for details. For symmetrical assemblies (all ranks use identical SDRAMs), must be coded as 0x00. | | |

| SDRAM Refresh Management (RFM), Second Byte<br>Byte 57 (Second SDRAM) | | |
|---|---|---|
| Bits 7~6<br>(MR59:OP[7:6]) | Bits 5~4<br>(MR59:OP[5:4]) | Bits 3~0<br>(MR59:OP[3:0]) |
| RFM RAA Counter Decrement per REF command | ARFM Level | Reserved |
| See byte 55 for details. For symmetrical assemblies (all ranks use identical SDRAMs), must be coded as 0x00. | | |

## 8.21  Bytes 58~69 (0x03A~0x045): SDRAM Adaptive Refresh Management

These bytes represent the adaptive refresh management (ARFM) settings for the SDRAMs used on this module. These values are vendor specific and come from the vendor's DDR5 SDRAM data sheet.

| SDRAM Adaptive Refresh Management (ARFM) | | | | |
|---|---|---|---|---|
| ARFM Level | First SDRAM | | Second SDRAM | |
| | First Byte | Second Byte | First Byte | Second Byte |
| A | SPD Byte 58 (0x03A) | SPD Byte 59 (0x03B) | SPD Byte 60 (0x03C) | SPD Byte 61 (0x03D) |
| B | SPD Byte 62 (0x03E) | SPD Byte 63 (0x03F) | SPD Byte 64 (0x040) | SPD Byte 65 (0x041) |
| C | SPD Byte 66 (0x042) | SPD Byte 67 (0x043) | SPD Byte 68 (0x044) | SPD Byte 69 (0x045) |
| NOTES: <br> ARFM = Adaptive Refresh Management <br> These bytes use similar encoding to the RFM bytes 54~55, with the difference that bit 0 of all First Bytes is to be decoded as "ARFM Level Supported" | | | | |

| SDRAM Adaptive Refresh Management (ARFM), First Bytes | | |
|---|---|---|
| Bits 7~5 <br> (MR58:OP[7:5]) | Bits 4~1 <br> (MR58:OP[4:1]) | Bit 0 <br> (MR58:OP[0]) |
| RAAMMT | RAAIMT | ARFM Level |
| See byte 54 for details. | | 0: ARFM Level not supported <br> 1: ARFM Level supported |

| SDRAM Adaptive Refresh Management (ARFM), Second Bytes | | |
|---|---|---|
| Bits 7~6 <br> (MR59:OP[7:6]) | Bits 5~4 <br> (MR59:OP[5:5]) | Bits 3~0 <br> (MR59:OP[3~0]) |
| RFM RAA Counter Decrement per REF command | ARFM Level | Reserved |
| See byte 55 for details. | | |

## 8.22  SDRAM Minimum Active to Active Command Delay Time, Same Bank Group (t$_{RRD\_L}$min)

### 8.22.1  Byte 70 (0x046): Least Significant Byte
### Byte 71 (0x047): Most Significant Byte
### Byte 72 (0x048): Lower Clock Limit

The 16-bit word in the lower two bytes defines the minimum SDRAM Active to Active Delay Time in picoseconds (ps). The third byte defines the minimum number of clocks (nCK) required. For monolithic SDRAMs, this parameter is tRRD_L(1K) min or tRRD_L(2K) min; for 3DS devices, this parameter is tRRD_L_slr (1K) min. These values come from the DDR5 SDRAM data sheet, JESD79-5. The general algorithm is:

- Calculate the target tRRD_L clock count by applying the standard rounding algorithm to the 16-bit ps value and the application clock period in ps.
- Set the application tRRD_L in nCK units to the greater of the target tRRD_L or the Lower Clock Limit.

| SDRAM Minimum Active to Active Command Delay Time (t$_{RRD\_L}$min) | | |
|---|---|---|
| nCK | ps | |
| Byte 72 | Byte 71 | Byte 70 |
| Bits 7~0 | Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 255 clocks (nCK) 0x00: Reserved | Values defined from 1 to 65535 ps 0x0000: Reserved | |

## 8.23  SDRAM Minimum Read to Read Command Delay Time, Same Bank Group (t$_{CCD\_L}$min)

### 8.23.1  Byte 73 (0x049): Least Significant Byte
### Byte 74 (0x04A): Most Significant Byte
### Byte 75 (0x04B): Lower Clock Limit

The 16-bit word in the lower two bytes defines the minimum SDRAM Read to Read Command Delay Time in picoseconds (ps) when accessing the same bank group. The third byte defines the minimum number of clocks (nCK) required. For monolithic SDRAMs, this parameter is tCCD_L min; for 3DS devices, this parameter is tCCD_L_slr min. These values come from the DDR5 SDRAM data sheet, JESD79-5. The general algorithm is:

- Calculate the target tCCD_L clock count by applying the standard rounding algorithm to the 16-bit ps value and the application clock period in ps.
- Set the application tCCD_L in nCK units to the greater of the target tCCD_L or the lower clock limit.

| SDRAM Minimum CAS_n to CAS_n Command Delay Time (t$_{CCD\_L}$min) | | |
|---|---|---|
| nCK | ps | |
| Byte 75 | Byte 74 | Byte 73 |
| Bits 7~0 | Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 255 clocks (nCK) 0x00: Reserved | Values defined from 1 to 65535 ps 0x0000: Reserved | |

## 8.24 SDRAM Minimum Write to Write Command Delay Time, Same Bank Group ($t_{CCD\_L\_WR}$min)

### 8.24.1 Byte 76 (0x04C): Least Significant Byte
Byte 77 (0x04D): Most Significant Byte
Byte 78 (0x04E): Lower Clock Limit

The 16-bit word in the lower two bytes defines the minimum SDRAM Write to Write Command Delay Time in picoseconds (ps) when accessing the same bank group. The third byte defines the minimum number of clocks (nCK) required. For monolithic SDRAMs, this parameter is tCCD_L_WR min; for 3DS devices, this parameter is tCCD_L_WR_slr min. These values come from the DDR5 SDRAM data sheet, JESD79-5. The general algorithm is:

- Calculate the target tCCD_L_WR clock count by applying the standard rounding algorithm to the 16-bit ps value and the application clock period in ps.
- Set the application tCCD_L_WR in nCK units to the greater of the target tCCD_L_WR or the lower clock limit.

| SDRAM Minimum Write to Write Command Delay Time ($t_{CCD\_L\_WR}$min) | | |
|---|---|---|
| nCK | ps | |
| Byte 78 | Byte 77 | Byte 76 |
| Bits 7~0 | Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 255 clocks (nCK)<br>0x00: Reserved | Values defined from 1 to 65535 ps<br>0x0000: Reserved | |

### 8.25 SDRAM Minimum Write to Write Command Delay Time, Second Write not RMW, Same Bank Group ($t_{CCD\_L\_WR2}$min)

#### 8.25.1 Byte 79 (0x04F): Least Significant Byte
Byte 80 (0x050): Most Significant Byte
Byte 81 (0x051): Lower Clock Limit

The 16-bit word in the lower two bytes defines the minimum SDRAM Write to Write Command Delay Time in picoseconds (ps) when accessing the same bank group and the second write is not a read-modify-write. The third byte defines the minimum number of clocks (nCK) required. For monolithic SDRAMs, this parameter is tCCD_L_WR2 min; for 3DS devices, this parameter is tCCD_L_WR2_slr min. These values come from the DDR5 SDRAM data sheet, JESD79-5. The general algorithm is:

- Calculate the target tCCD_L_WR2 clock count by applying the standard rounding algorithm to the 16-bit ps value and the application clock period in ps.
- Set the application tCCD_L_WR2 in nCK units to the greater of the target tCCD_L_WR2 or the lower clock limit.

| SDRAM Minimum Write to Write Command Delay Time, Second Write not RMW ($t_{CCD\_L\_WR2}$min) | | |
|---|---|---|
| nCK | ps | |
| Byte 81 | Byte 80 | Byte 79 |
| Bits 7~0 | Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 255 clocks (nCK)<br>0x00: Reserved | Values defined from 1 to 65535 ps<br>0x0000: Reserved | |

### 8.26 SDRAM Minimum Four Activate Window ($t_{FAW}$min)

#### 8.26.1 Byte 82 (0x052): Least Significant Byte
Byte 83 (0x053): Most Significant Byte
Byte 84 (0x054): Lower Clock Limit

The 16-bit word in the lower two bytes defines the minimum SDRAM Four Activate Window in picoseconds (ps). The third byte defines the minimum number of clocks (nCK) required. For monolithic SDRAMs, this parameter is tFAW(1K) min or tFAW(2K) min; for 3DS devices, this parameter is tFAW_slr (1K) min. These values come from the DDR5 SDRAM data sheet, JESD79-5. The general algorithm is:

- Calculate the target tFAW clock count by applying the standard rounding algorithm to the 16-bit ps value and the application clock period in ps.
- Set the application tFAW in nCK units to the greater of the target tFAW or the lower clock limit.

| SDRAM Minimum Four Activate Window ($t_{FAW}$min) | | |
|---|---|---|
| nCK | ps | |
| Byte 84 | Byte 83 | Byte 82 |
| Bits 7~0 | Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 255 clocks (nCK)<br>0x00: Reserved | Values defined from 1 to 65535 ps<br>0x0000: Reserved | |

### 8.27 SDRAM Minimum Write to Read Command Delay Time, Same Bank Group ($t_{CCD\_L\_WTR}min$)

#### 8.27.1 Byte 85 (0x055): Least Significant Byte
Byte 86 (0x056): Most Significant Byte
Byte 87 (0x057): Lower Clock Limit

The 16-bit word in the lower two bytes defines the minimum Write to Read Command Delay Time in picoseconds (ps) when accessing the same bank group. The third byte defines the minimum number of clocks (nCK) required. For monolithic SDRAMs, this parameter is tCCD_L_WTR min; for 3DS devices, this parameter is tCCD_L_WTR_slr min. These values come from the DDR5 SDRAM data sheet, JESD79-5. The general algorithm is:

- Calculate the command time in clocks for CAS Write Latency (CWL) plus Write Burst Length divided by 2.
- Calculate the target WTR(L) clock count by applying the standard rounding algorithm to the 16-bit ps value and the application clock period in ps.
- Set the application WTR(L) in nCK units to the greater of the target WTR(L) or the lower clock limit.
- Set the application tCCD_L_WTR to the sum of command time and WTR(L): tCCD_L_WTR = CWL + WBL/2 + WTR(L)

| SDRAM Minimum Write to Read Command Delay Time ($t_{CCD\_L\_WTR}min$), Same Bank Group | | |
|---|---|---|
| nCK | ps | |
| Byte 87 | Byte 86 | Byte 85 |
| Bits 7~0 | Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 255 clocks (nCK) 0x00: Reserved | Values defined from 1 to 65535 ps 0x0000: Reserved | |

### 8.28 SDRAM Minimum Write to Read Command Delay Time, Different Bank Group ($t_{CCD\_S\_WTR}min$)

#### 8.28.1 Byte 88 (0x058): Least Significant Byte
Byte 89 (0x059): Most Significant Byte
Byte 90 (0x05A): Lower Clock Limit

The 16-bit word in the lower two bytes defines the minimum Write to Read Command Delay Time in picoseconds (ps) when accessing different bank groups. The third byte defines the minimum number of clocks (nCK) required. For monolithic SDRAMs, this parameter is tWTR_S min; for 3DS devices, this parameter is tWTR_S_slr min. These values come from the DDR5 SDRAM data sheet, JESD79-5. The general algorithm is:

- Calculate the command time in clocks for CAS Write Latency (CWL) plus Write Burst Length divided by 2.
- Calculate the target WTR(S) clock count by applying the standard rounding algorithm to the 16-bit ps value and the application clock period in ps.
- Set the application WTR(S) in nCK units to the greater of the target WTR(S) or the lower clock limit.
- Set the application tCCD_L_WTR to the sum of command time and WTR(S): tCCD_S_WTR = CWL + WBL/2 + WTR(S)

| SDRAM Minimum Write to Read Command Delay Time ($t_{CCD\_S\_WTR}min$), Different Bank Group | | |
|---|---|---|
| nCK | ps | |
| Byte 90 | Byte 89 | Byte 89 |
| Bits 7~0 | Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 255 clocks (nCK) 0x00: Reserved | Values defined from 1 to 65535 ps 0x0000: Reserved | |

### 8.29 SDRAM Minimum Read to Precharge Command Delay Time, ($t_{RTP}$min)

### 8.29.1 Byte 91 (0x05B): Least Significant Byte
### Byte 92 (0x05C): Most Significant Byte
### Byte 93 (0x05D): Lower Clock Limit

The 16-bit word in the lower two bytes defines the minimum internal read to precharge Command Delay Time in picoseconds (ps) when in different bank groups. The third byte defines the minimum number of clocks (nCK) required. For monolithic SDRAMs, this parameter is tRTP min; for 3DS devices, this parameter is tRTP_slr min. These values come from the DDR5 SDRAM data sheet, JESD79-5. The general algorithm is:

- Calculate the target tRTP clock count by applying the standard rounding algorithm to the 16-bit ps value and the application clock period in ps.
- Set the application tRTP in nCK units to the greater of the target tRTP or the lower clock limit.

Parameter tWTRA (internal write to read with autoprecharge) is calculated as tWTRA = tWR - tRTP. This calculation must be maintained when using the rounding algorithm on each parameter individually, or rounding on the result of the subtraction, and the higher number of resulting clocks must be used.

| SDRAM Minimum Write to Read Command Delay Time ($t_{RTP}$min), Different Bank Group | | |
|---|---|---|
| nCK | ps | |
| Byte 87 | Byte 86 | Byte 85 |
| Bits 7~0 | Bits 15~8 | Bits 7~0 |
| Values defined from 1 to 255 clocks (nCK) 0x00: Reserved | Values defined from 1 to 65535 ps 0x0000: Reserved | |

### 8.30 Bytes 94~127 (0x05E~0x07F): Reserved, Base Configuration Section

Must be coded as 0x00

# 9    Example SPD Codes

Tables 6 through 8 show example codes for the SPD contents. These calculations account for the 0.3% correction factor allowed for digital math inaccuracies. The clock periods for each standard speed grade used for these calculations are:

| tCK | | |
|---|---|---|
| Grade | tCKmin | tCKmax |
| 3200 | 0.625 | 1.010 |
| 3600 | 0.555 | 1.010 |
| 4000 | 0.500 | 1.010 |
| 4400 | 0.454 | 1.010 |
| 4800 | 0.416 | 1.010 |
| 5200 | 0.384 | 1.010 |
| 5600 | 0.357 | 1.010 |
| 6000 | 0.333 | 1.010 |
| 6400 | 0.312 | 1.010 |

## 9.1    Core parameters, DDR5 Monolithic Single Die Package (SDP) SDRAMs

### Table 6 — Example #1, Codes for SPD Contents

| | | 3200AN Mono | 3200B Mono | 3200BN Mono | 3200C Mono | 3600AN Mono | 3600B Mono | 3600BN Mono | 3600C Mono | 4000AN Mono | 4000B Mono | 4000BN Mono | 4000C Mono |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | tCKmin | 0.625 | 0.625 | 0.625 | 0.625 | 0.555 | 0.555 | 0.555 | 0.555 | 0.500 | 0.500 | 0.500 | 0.500 |
| | tCKmax | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 |
| | tAA | 15.000 | 16.250 | 16.250 | 17.500 | 14.444 | 16.250 | 16.666 | 17.500 | 14.000 | 16.000 | 16.000 | 17.500 |
| | tRCD | 15.000 | 16.250 | 16.250 | 17.500 | 14.444 | 16.250 | 16.666 | 17.500 | 14.000 | 16.000 | 16.000 | 17.500 |
| | tRP | 15.000 | 16.250 | 16.250 | 17.500 | 14.444 | 16.250 | 16.666 | 17.500 | 14.000 | 16.000 | 16.000 | 17.500 |
| | tRAS | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 |
| | tRC | 47.000 | 48.250 | 48.250 | 49.500 | 46.444 | 46.250 | 48.666 | 49.500 | 46.000 | 48.000 | 48.000 | 49.500 |
| | tWR | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 |
| SPD # | tCKmin (ps) | 625 | 625 | 625 | 625 | 555 | 555 | 555 | 555 | 500 | 500 | 500 | 500 |
| 20 | tCKmin low | 0x71 | 0x71 | 0x71 | 0x71 | 0x2B | 0x2B | 0x2B | 0x2B | 0xF4 | 0xF4 | 0xF4 | 0xF4 |
| 21 | tCKmin high | 0x02 | 0x02 | 0x02 | 0x02 | 0x02 | 0x02 | 0x02 | 0x02 | 0x01 | 0x01 | 0x01 | 0x01 |
| SPD # | tCKmax (ps) | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 |
| 22 | tCKmax low | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 |
| 23 | tCKmax high | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 |
| SPD # | tAA (ps) | 15000 | 16250 | 16250 | 17500 | 14444 | 16250 | 16666 | 17500 | 14000 | 16000 | 16000 | 17500 |
| 30 | tAA low | 0x98 | 0x7A | 0x7A | 0x5C | 0x6C | 0x7A | 0x1A | 0x5C | 0xB0 | 0x80 | 0x80 | 0x5C |
| 31 | tAA high | 0x3A | 0x3F | 0x3F | 0x44 | 0x38 | 0x3F | 0x41 | 0x44 | 0x36 | 0x3E | 0x3E | 0x44 |
| SPD # | tRCD (ps) | 15000 | 16250 | 16250 | 17500 | 14444 | 16250 | 16666 | 17500 | 14000 | 16000 | 16000 | 17500 |
| 32 | tRCD low | 0x98 | 0x7A | 0x7A | 0x5C | 0x6C | 0x7A | 0x1A | 0x5C | 0xB0 | 0x80 | 0x80 | 0x5C |
| 33 | tRCD high | 0x3A | 0x3F | 0x3F | 0x44 | 0x38 | 0x3F | 0x41 | 0x44 | 0x36 | 0x3E | 0x3E | 0x44 |
| SPD # | tRP (ps) | 15000 | 16250 | 16250 | 17500 | 14444 | 16250 | 16666 | 17500 | 14000 | 16000 | 16000 | 17500 |
| 34 | tRP low | 0x98 | 0x7A | 0x7A | 0x5C | 0x6C | 0x7A | 0x1A | 0x5C | 0xB0 | 0x80 | 0x80 | 0x5C |
| 35 | tRP high | 0x3A | 0x3F | 0x3F | 0x44 | 0x38 | 0x3F | 0x41 | 0x44 | 0x36 | 0x3E | 0x3E | 0x44 |
| SPD # | tRAS (ps) | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 |
| 36 | tRAS low | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |
| 37 | tRAS high | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D |
| SPD # | tRC (ps) | 47000 | 48250 | 48250 | 49500 | 46444 | 46250 | 48666 | 49500 | 46000 | 48000 | 48000 | 49500 |
| 38 | tRC low | 0x98 | 0x7A | 0x7A | 0x5C | 0x6C | 0xAA | 0x1A | 0x5C | 0xB0 | 0x80 | 0x80 | 0x5C |
| 39 | tRC high | 0xB7 | 0xBC | 0xBC | 0xC1 | 0xB5 | 0xB4 | 0xBE | 0xC1 | 0xB3 | 0xBB | 0xBB | 0xC1 |
| SPD # | tWR (ps) | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 |
| 40 | tWR low | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 |
| 41 | tWR high | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 |

## 9.1 Core parameters, DDR5 Monolithic Single Die Package (SDP) SDRAMs (cont'd)

**Table 7 — Example #2, Codes for SPD Contents**

| | | 4400AN Mono | 4400B Mono | 4400BN Mono | 4400C Mono | 4800AN Mono | 4800B Mono | 4800BN Mono | 4800C Mono | 5200AN Mono | 5200B Mono | 5200BN Mono | 5200C Mono |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | tCKmin | 0.454 | 0.454 | 0.454 | 0.454 | 0.416 | 0.416 | 0.416 | 0.416 | 0.384 | 0.384 | 0.384 | 0.384 |
| | tCKmax | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 |
| | tAA | 14.545 | 16.000 | 16.363 | 17.500 | 14.166 | 16.000 | 16.666 | 17.500 | 14.615 | 16.000 | 16.153 | 17.500 |
| | tRCD | 14.545 | 16.000 | 16.363 | 17.500 | 14.166 | 16.000 | 16.666 | 17.500 | 14.615 | 16.000 | 16.153 | 17.500 |
| | tRP | 14.545 | 16.000 | 16.363 | 17.500 | 14.166 | 16.000 | 16.666 | 17.500 | 14.615 | 16.000 | 16.153 | 17.500 |
| | tRAS | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 |
| | tRC | 46.545 | 48.000 | 48.363 | 49.500 | 46.166 | 48.000 | 48.666 | 49.500 | 46.615 | 48.000 | 48.153 | 49.500 |
| | tWR | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 |
| SPD # | tCKmin (ps) | 454 | 454 | 454 | 454 | 416 | 416 | 416 | 416 | 384 | 384 | 384 | 384 |
| 20 | tCKmin low | 0xC6 | 0xC6 | 0xC6 | 0xC6 | 0xA0 | 0xA0 | 0xA0 | 0xA0 | 0x80 | 0x80 | 0x80 | 0x80 |
| 21 | tCKmin high | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 |
| SPD # | tCKmax (ps) | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 |
| 22 | tCKmax low | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 |
| 23 | tCKmax high | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 |
| SPD # | tAA (ps) | 14545 | 16000 | 16363 | 17500 | 14166 | 16000 | 16666 | 17500 | 14615 | 16000 | 16153 | 17500 |
| 30 | tAA low | 0xD1 | 0x80 | 0xEB | 0x5C | 0x56 | 0x80 | 0x1A | 0x5C | 0x17 | 0x80 | 0x18 | 0x5C |
| 31 | tAA high | 0x38 | 0x3E | 0x3F | 0x44 | 0x37 | 0x3E | 0x41 | 0x44 | 0x39 | 0x3E | 0x3F | 0x44 |
| SPD # | tRCD (ps) | 14545 | 16000 | 16363 | 17500 | 14166 | 16000 | 16666 | 17500 | 14615 | 16000 | 16153 | 17500 |
| 32 | tRCD low | 0xD1 | 0x80 | 0xEB | 0x5C | 0x56 | 0x80 | 0x1A | 0x5C | 0x17 | 0x80 | 0x18 | 0x5C |
| 33 | tRCD high | 0x38 | 0x3E | 0x3F | 0x44 | 0x37 | 0x3E | 0x41 | 0x44 | 0x39 | 0x3E | 0x3F | 0x44 |
| SPD # | tRP (ps) | 14545 | 16000 | 16363 | 17500 | 14166 | 16000 | 16666 | 17500 | 14615 | 16000 | 16153 | 17500 |
| 34 | tRP low | 0xD1 | 0x80 | 0xEB | 0x5C | 0x56 | 0x80 | 0x1A | 0x5C | 0x17 | 0x80 | 0x18 | 0x5C |
| 35 | tRP high | 0x38 | 0x3E | 0x3F | 0x44 | 0x37 | 0x3E | 0x41 | 0x44 | 0x39 | 0x3E | 0x3F | 0x44 |
| SPD # | tRAS (ps) | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 |
| 36 | tRAS low | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |
| 37 | tRAS high | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D |
| SPD # | tRC (ps) | 46545 | 48000 | 48363 | 49500 | 46166 | 48000 | 48666 | 49500 | 46615 | 48000 | 48153 | 49500 |
| 38 | tRC low | 0xD1 | 0x80 | 0xEB | 0x5C | 0x56 | 0x80 | 0x1A | 0x5C | 0x17 | 0x80 | 0x19 | 0x5C |
| 39 | tRC high | 0xB5 | 0xBB | 0xBC | 0xC1 | 0xB4 | 0xBB | 0xBE | 0xC1 | 0xB6 | 0xBB | 0xBC | 0xC1 |
| SPD # | tWR (ps) | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 |
| 40 | tWR low | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 |
| 41 | tWR high | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 |

## 9.1　Core parameters, DDR5 Monolithic Single Die Package (SDP) SDRAMs (cont'd)

### Table 8 — Example #3, Codes for SPD Contents

| | | 5600AN Mono | 5600B Mono | 5600BN Mono | 5600C Mono | 6000AN Mono | 6000B Mono | 6000BN Mono | 6000C Mono | 6400AN Mono | 6400B Mono | 6400BN Mono | 6400C Mono |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | tCKmin | 0.357 | 0.357 | 0.357 | 0.357 | 0.333 | 0.333 | 0.333 | 0.333 | 0.312 | 0.312 | 0.312 | 0.312 |
| | tCKmax | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 |
| | tAA | 14.285 | 16.000 | 16.428 | 17.500 | 14.000 | 16.000 | 16.000 | 17.500 | 14.375 | 16.000 | 16.250 | 17.500 |
| | tRCD | 14.285 | 16.000 | 16.428 | 17.500 | 14.000 | 16.000 | 16.000 | 17.500 | 14.375 | 16.000 | 16.250 | 17.500 |
| | tRP | 14.285 | 16.000 | 16.428 | 17.500 | 14.000 | 16.000 | 16.000 | 17.500 | 14.375 | 16.000 | 16.250 | 17.500 |
| | tRAS | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 |
| | tRC | 46.285 | 48.000 | 48.428 | 49.500 | 46.000 | 48.000 | 48.000 | 49.500 | 46.375 | 48.000 | 48.250 | 49.500 |
| | tWR | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 |
| SPD # | tCKmin (ps) | 357 | 357 | 357 | 357 | 333 | 333 | 333 | 333 | 312 | 312 | 312 | 312 |
| 20 | tCKmin low | 0x65 | 0x65 | 0x65 | 0x65 | 0x4D | 0x4D | 0x4D | 0x4D | 0x38 | 0x38 | 0x38 | 0x38 |
| 21 | tCKmin high | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 |
| SPD # | tCKmax (ps) | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 |
| 22 | tCKmax low | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 |
| 23 | tCKmax high | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 |
| SPD # | tAA (ps) | 14285 | 16000 | 16428 | 17500 | 14000 | 16000 | 16000 | 17500 | 14375 | 16000 | 16250 | 17500 |
| 30 | tAA low | 0xCD | 0x80 | 0x2C | 0x5C | 0xB0 | 0x80 | 0x80 | 0x5C | 0x27 | 0x80 | 0x7A | 0x5C |
| 31 | tAA high | 0x37 | 0x3E | 0x40 | 0x44 | 0x36 | 0x3E | 0x3E | 0x44 | 0x38 | 0x3E | 0x3F | 0x44 |
| SPD # | tRCD (ps) | 14285 | 16000 | 16428 | 17500 | 14000 | 16000 | 16000 | 17500 | 14375 | 16000 | 16250 | 17500 |
| 32 | tRCD low | 0xCD | 0x80 | 0x2C | 0x5C | 0xB0 | 0x80 | 0x80 | 0x5C | 0x27 | 0x80 | 0x7A | 0x5C |
| 33 | tRCD high | 0x37 | 0x3E | 0x40 | 0x44 | 0x36 | 0x3E | 0x3E | 0x44 | 0x38 | 0x3E | 0x3F | 0x44 |
| SPD # | tRP (ps) | 14285 | 16000 | 16428 | 17500 | 14000 | 16000 | 16000 | 17500 | 14375 | 16000 | 16250 | 17500 |
| 34 | tRP low | 0xCD | 0x80 | 0x2C | 0x5C | 0xB0 | 0x80 | 0x80 | 0x5C | 0x27 | 0x80 | 0x7A | 0x5C |
| 35 | tRP high | 0x37 | 0x3E | 0x40 | 0x44 | 0x36 | 0x3E | 0x3E | 0x44 | 0x38 | 0x3E | 0x3F | 0x44 |
| SPD # | tRAS (ps) | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 |
| 36 | tRAS low | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |
| 37 | tRAS high | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D |
| SPD # | tRC (ps) | 46285 | 48000 | 48428 | 49500 | 46000 | 48000 | 48000 | 49500 | 46375 | 48000 | 48250 | 49500 |
| 38 | tRC low | 0xCD | 0x80 | 0x2C | 0x5C | 0xB0 | 0x80 | 0x80 | 0x5C | 0x27 | 0x80 | 0x7A | 0x5C |
| 39 | tRC high | 0xB4 | 0xBB | 0xBD | 0xC1 | 0xB3 | 0xBB | 0xBB | 0xC1 | 0xB5 | 0xBB | 0xBC | 0xC1 |
| SPD # | tWR (ps) | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 |
| 40 | tWR low | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 |
| 41 | tWR high | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 |

## 9.2    Parameters with Lower nCK Limits, DDR5 Monolithic Single Die Package SDRAMs

| | | DDR5-3200 Mono | DDR5-3600 Mono | DDR5-4000 Mono | DDR5-4400 Mono | DDR5-4800 Mono | DDR5-5200 Mono | DDR5-5600 Mono | DDR5-6000 Mono | DDR5-6400 Mono |
|---|---|---|---|---|---|---|---|---|---|---|
| tRRD_L (2K) | ps | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 |
| tRRD_L (2K) | nCK | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| tRRD_L (1K) | ps | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 |
| tRRD_L (1K) | nCK | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| tCCD_L | ps | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 |
| tCCD_L | nCK | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| tCCD_L_WR | ps | 20000 | 20000 | 20000 | 20000 | 20000 | 20000 | 20000 | 20000 | 20000 |
| tCCD_L_WR | nCK | 32 | 32 | 32 | 32 | 32 | 32 | 32 | 32 | 32 |
| tCCD_L_WR2 | ps | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 |
| tCCD_L_WR2 | nCK | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| tFAW (2K) | ps | 25000 | 22222 | 20000 | 18181 | 16666 | 15384 | 14285 | 13333 | 12500 |
| tFAW (2K) | nCK | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 |
| tFAW (1K) | ps | 20000 | 17777 | 16000 | 14545 | 13333 | 12307 | 11428 | 10666 | 10000 |
| tFAW (1K) | nCK | 32 | 32 | 32 | 32 | 32 | 32 | 32 | 32 | 32 |
| tCCD_L_WTR | ps | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 |
| tCCD_L_WTR | nCK | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| tCCD_S_WTR | ps | 2500 | 2500 | 2500 | 2500 | 2500 | 2500 | 2500 | 2500 | 2500 |
| tCCD_S_WTR | nCK | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| tRTP | ps | 7500 | 7500 | 7500 | 7500 | 7500 | 7500 | 7500 | 7500 | 7500 |
| tRTP | nCK | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 |

| SPD# | | | DDR5-3200 Mono | DDR5-3600 Mono | DDR5-4000 Mono | DDR5-4400 Mono | DDR5-4800 Mono | DDR5-5200 Mono | DDR5-5600 Mono | DDR5-6000 Mono | DDR5-6400 Mono |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 71, 70 | tRRD_L (2K) | MSB, LSB | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 |
| 72 | tRRD_L (2K) | nCK | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 |
| 71, 70 | tRRD_L (1K) | MSB, LSB | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 |
| 72 | tRRD_L (1K) | nCK | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 |
| 74, 73 | tCCD_L | MSB, LSB | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 |
| 75 | tCCD_L | nCK | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 |
| 77, 76 | tCCD_L_WR | MSB, LSB | 0x4E, 0x20 | 0x4E, 0x20 | 0x4E, 0x20 | 0x4E, 0x20 | 0x4E, 0x20 | 0x4E, 0x20 | 0x4E, 0x20 | 0x4E, 0x20 | 0x4E, 0x20 |
| 78 | tCCD_L_WR | nCK | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 |
| 80, 79 | tCCD_L_WR2 | MSB, LSB | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 |
| 81 | tCCD_L_WR2 | nCK | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 |
| 83, 82 | tFAW (2K) | MSB, LSB | 0x61, 0xA8 | 0x56, 0xCE | 0x4E, 0x20 | 0x47, 0x05 | 0x41, 0x1A | 0x3C, 0x18 | 0x37, 0xCD | 0x34, 0x15 | 0x30, 0xD4 |
| 84 | tFAW (2K) | nCK | 0x28 | 0x28 | 0x28 | 0x28 | 0x28 | 0x28 | 0x28 | 0x28 | 0x28 |
| 83, 82 | tFAW (1K) | MSB, LSB | 0x4E, 0x20 | 0x45, 0x71 | 0x3E, 0x80 | 0x38, 0xD1 | 0x34, 0x15 | 0x30, 0x13 | 0x2C, 0xA4 | 0x29, 0xAA | 0x27, 0x10 |
| 84 | tFAW (1K) | nCK | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 |
| 86, 85 | tCCD_L_WTR | MSB, LSB | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 |
| 87 | tCCD_L_WTR | nCK | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 |
| 89, 88 | tCCD_S_WTR | MSB, LSB | 0x09, 0xC4 | 0x09, 0xC4 | 0x09, 0xC4 | 0x09, 0xC4 | 0x09, 0xC4 | 0x09, 0xC4 | 0x09, 0xC4 | 0x09, 0xC4 | 0x09, 0xC4 |
| 90 | tCCD_S_WTR | nCK | 0x04 | 0x04 | 0x04 | 0x04 | 0x04 | 0x04 | 0x04 | 0x04 | 0x04 |
| 92, 91 | tRTP | MSB, LSB | 0x1D, 0x4C | 0x1D, 0x4C | 0x1D, 0x4C | 0x1D, 0x4C | 0x1D, 0x4C | 0x1D, 0x4C | 0x1D, 0x4C | 0x1D, 0x4C | 0x1D, 0x4C |
| 93 | tRTP | nCK | 0x0C | 0x0C | 0x0C | 0x0C | 0x0C | 0x0C | 0x0C | 0x0C | 0x0C |

## 9.3    CAS Latency Masks, DDR5 Monolithic Single Die Package SDRAMs

Optional downbins are included. Optional CAS latencies are not included.

| Mono Speed Bin | SPD[24] | SPD[25] | SPD[26] | SPD[27] | SPD[28] |
|---|---|---|---|---|---|
| 3200AN | 0x1E | 0x00 | 0x00 | 0x00 | 0x00 |
| 3200B | 0x1A | 0x00 | 0x00 | 0x00 | 0x00 |
| 3200BN | 0x1A | 0x00 | 0x00 | 0x00 | 0x00 |
| 3200C | 0x12 | 0x00 | 0x00 | 0x00 | 0x00 |
| 3600AN | 0x7E | 0x00 | 0x00 | 0x00 | 0x00 |
| 3600B | 0x7A | 0x00 | 0x00 | 0x00 | 0x00 |
| 3600BN | 0x72 | 0x00 | 0x00 | 0x00 | 0x00 |
| 3600C | 0x52 | 0x00 | 0x00 | 0x00 | 0x00 |
| 4000AN | 0x7E | 0x01 | 0x00 | 0x00 | 0x00 |
| 4000B | 0x7A | 0x01 | 0x00 | 0x00 | 0x00 |
| 4000BN | 0x7A | 0x01 | 0x00 | 0x00 | 0x00 |
| 4000C | 0x52 | 0x01 | 0x00 | 0x00 | 0x00 |
| 4400AN | 0x7E | 0x05 | 0x00 | 0x00 | 0x00 |
| 4400B | 0x7A | 0x05 | 0x00 | 0x00 | 0x00 |
| 4400BN | 0x72 | 0x05 | 0x00 | 0x00 | 0x00 |
| 4400C | 0x52 | 0x05 | 0x00 | 0x00 | 0x00 |
| 4800AN | 0xFE | 0x0D | 0x00 | 0x00 | 0x00 |
| 4800B | 0x7A | 0x0D | 0x00 | 0x00 | 0x00 |
| 4800BN | 0x72 | 0x0D | 0x00 | 0x00 | 0x00 |
| 4800C | 0x52 | 0x0D | 0x00 | 0x00 | 0x00 |
| 5200AN | 0x7E | 0x2F | 0x00 | 0x00 | 0x00 |
| 5200B | 0x7A | 0x2D | 0x00 | 0x00 | 0x00 |
| 5200BN | 0x7A | 0x2D | 0x00 | 0x00 | 0x00 |
| 5200C | 0x52 | 0x2D | 0x00 | 0x00 | 0x00 |
| 5600AN | 0x7E | 0xAF | 0x00 | 0x00 | 0x00 |
| 5600B | 0x7A | 0xAD | 0x00 | 0x00 | 0x00 |
| 5600BN | 0x72 | 0xAD | 0x00 | 0x00 | 0x00 |
| 5600C | 0x52 | 0xAD | 0x00 | 0x00 | 0x00 |
| 6000AN | 0xFE | 0xEF | 0x02 | 0x00 | 0x00 |
| 6000B | 0x7A | 0xED | 0x02 | 0x00 | 0x00 |
| 6000BN | 0x7A | 0xED | 0x02 | 0x00 | 0x00 |
| 6000C | 0x52 | 0xAD | 0x02 | 0x00 | 0x00 |
| 6400AN | 0x7E | 0xEF | 0x07 | 0x00 | 0x00 |
| 6400B | 0x7A | 0xED | 0x07 | 0x00 | 0x00 |
| 6400BN | 0x7A | 0xED | 0x07 | 0x00 | 0x00 |
| 6400C | 0x52 | 0xAD | 0x06 | 0x00 | 0x00 |

## 9.4 Parameters Defined by Device Density, DDR5 Monolithic Single Die Package (SDP) SDRAMs

| | ns | SPD[43]<br>MSB | SPD[42]<br>LSB |
|---|---|---|---|
| tRFC1, 8 Gb | 195 | 0x00 | 0xC3 |
| tRFC1, 16 Gb | 295 | 0x01 | 0x27 |
| tRFC1, 24 Gb | 410 | 0x01 | 0x9A |
| tRFC1, 32 Gb | 410 | 0x01 | 0x9A |

| | ns | SPD[45]<br>MSB | SPD[44]<br>LSB |
|---|---|---|---|
| tRFC2, 8 Gb | 130 | 0x00 | 0x82 |
| tRFC2, 16 Gb | 160 | 0x00 | 0xA0 |
| tRFC2, 24 Gb | 220 | 0x00 | 0xDC |
| tRFC2, 32 Gb | 220 | 0x00 | 0xDC |

| | ns | SPD[47]<br>MSB | SPD[46]<br>LSB |
|---|---|---|---|
| tRFCsb, 8 Gb | 115 | 0x00 | 0x73 |
| tRFCsb, 16 Gb | 130 | 0x00 | 0x82 |
| tRFCsb, 24 Gb | 190 | 0x00 | 0xBE |
| tRFCsb, 32 Gb | 190 | 0x00 | 0xBE |

## 9.5    Core parameters, DDR5 Multi-Die Packages (3DS) SDRAMs

| | | 3200AN 3DS | 3200B 3DS | 3200BN 3DS | 3200C 3DS | 3600AN 3DS | 3600B 3DS | 3600BN 3DS | 3600C 3DS | 4000AN 3DS | 4000B 3DS | 4000BN 3DS | 4000C 3DS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | tCKmin | 0.625 | 0.625 | 0.625 | 0.625 | 0.555 | 0.555 | 0.555 | 0.555 | 0.500 | 0.500 | 0.500 | 0.500 |
| | tCKmax | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 |
| | tAA | 16.250 | 18.750 | 18.750 | 20.000 | 16.666 | 18.750 | 18.888 | 20.000 | 16.000 | 18.750 | 19.000 | 20.000 |
| | tRCD | 15.000 | 16.250 | 16.250 | 17.500 | 14.444 | 16.250 | 16.666 | 17.500 | 14.000 | 16.000 | 16.000 | 17.500 |
| | tRP | 15.000 | 16.250 | 16.250 | 17.500 | 14.444 | 16.250 | 16.666 | 17.500 | 14.000 | 16.000 | 16.000 | 17.500 |
| | tRAS | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 |
| | tRC | 45.750 | 48.250 | 48.250 | 49.500 | 46.444 | 48.250 | 48.666 | 49.500 | 46.000 | 48.000 | 48.000 | 49.500 |
| | tWR | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 |
| SPD # | tCKmin (ps) | 625 | 625 | 625 | 625 | 555 | 555 | 555 | 555 | 500 | 500 | 500 | 500 |
| 20 | tCKmin low | 0x71 | 0x71 | 0x71 | 0x71 | 0x2B | 0x2B | 0x2B | 0x2B | 0xF4 | 0xF4 | 0xF4 | 0xF4 |
| 21 | tCKmin high | 0x02 | 0x02 | 0x02 | 0x02 | 0x02 | 0x02 | 0x02 | 0x02 | 0x01 | 0x01 | 0x01 | 0x01 |
| SPD # | tCKmax (ps) | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 |
| 22 | tCKmax low | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 |
| 23 | tCKmax high | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 |
| SPD # | tAA (ps) | 16250 | 18750 | 18750 | 20000 | 16666 | 18750 | 18888 | 20000 | 16000 | 18750 | 19000 | 20000 |
| 30 | tAA low | 0x7A | 0x3E | 0x3E | 0x20 | 0x1A | 0x3E | 0xC8 | 0x20 | 0x80 | 0x3E | 0x38 | 0x20 |
| 31 | tAA high | 0x3F | 0x49 | 0x49 | 0x4E | 0x41 | 0x49 | 0x49 | 0x4E | 0x3E | 0x49 | 0x4A | 0x4E |
| SPD # | tRCD (ps) | 15000 | 16250 | 16250 | 17500 | 14444 | 16250 | 16666 | 17500 | 14000 | 16000 | 16000 | 17500 |
| 32 | tRCD low | 0x98 | 0x7A | 0x7A | 0x5C | 0x6C | 0x7A | 0x1A | 0x5C | 0xB0 | 0x80 | 0x80 | 0x5C |
| 33 | tRCD high | 0x3A | 0x3F | 0x3F | 0x44 | 0x38 | 0x3F | 0x41 | 0x44 | 0x36 | 0x3E | 0x3E | 0x44 |
| SPD # | tRP (ps) | 15000 | 16250 | 16250 | 17500 | 14444 | 16250 | 16666 | 17500 | 14000 | 16000 | 16000 | 17500 |
| 34 | tRP low | 0x98 | 0x7A | 0x7A | 0x5C | 0x6C | 0x7A | 0x1A | 0x5C | 0xB0 | 0x80 | 0x80 | 0x5C |
| 35 | tRP high | 0x3A | 0x3F | 0x3F | 0x44 | 0x38 | 0x3F | 0x41 | 0x44 | 0x36 | 0x3E | 0x3E | 0x44 |
| SPD # | tRAS (ps) | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 |
| 36 | tRAS low | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |
| 37 | tRAS high | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D |
| SPD # | tRC (ps) | 45750 | 48250 | 48250 | 49500 | 46444 | 48250 | 48666 | 49500 | 46000 | 48000 | 48000 | 49500 |
| 38 | tRC low | 0xB6 | 0x7A | 0x7A | 0x5C | 0x6C | 0x7A | 0x1A | 0x5C | 0xB0 | 0x80 | 0x80 | 0x5C |
| 39 | tRC high | 0xB2 | 0xBC | 0xBC | 0xC1 | 0xB5 | 0xBC | 0xBE | 0xC1 | 0xB3 | 0xBB | 0xBB | 0xC1 |
| SPD # | tWR (ps) | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 |
| 40 | tWR low | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 |
| 41 | tWR high | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 |

## 9.5 Core parameters, DDR5 Multi-Die Packages (3DS) SDRAMs (cont'd)

| | | 4400AN 3DS | 4400B 3DS | 4400BN 3DS | 4400C 3DS | 4800AN 3DS | 4800B 3DS | 4800BN 3DS | 4800C 3DS | 5200AN 3DS | 5200B 3DS | 5200BN 3DS | 5200C 3DS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| tCKmin | | 0.454 | 0.454 | 0.454 | 0.454 | 0.416 | 0.416 | 0.416 | 0.416 | 0.384 | 0.384 | 0.384 | 0.384 |
| tCKmax | | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 |
| tAA | | 16.363 | 18.750 | 19.090 | 20.000 | 16.666 | 18.750 | 19.166 | 20.000 | 16.153 | 18.750 | 19.230 | 20.000 |
| tRCD | | 14.545 | 16.000 | 16.363 | 17.500 | 14.166 | 16.000 | 16.666 | 17.500 | 14.615 | 16.000 | 16.153 | 17.500 |
| tRP | | 14.545 | 16.000 | 16.363 | 17.500 | 14.166 | 16.000 | 16.666 | 17.500 | 14.615 | 16.000 | 16.153 | 17.500 |
| tRAS | | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 |
| tRC | | 46.545 | 48.000 | 48.363 | 49.500 | 46.166 | 48.000 | 48.666 | 49.500 | 46.615 | 48.000 | 49.500 | 49.500 |
| tWR | | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 |
| SPD # | tCKmin (ps) | 454 | 454 | 454 | 454 | 416 | 416 | 416 | 416 | 384 | 384 | 384 | 384 |
| 20 | tCKmin low | 0xC6 | 0xC6 | 0xC6 | 0xC6 | 0xA0 | 0xA0 | 0xA0 | 0xA0 | 0x80 | 0x80 | 0x80 | 0x80 |
| 21 | tCKmin high | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 |
| SPD # | tCKmax (ps) | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 |
| 22 | tCKmax low | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 |
| 23 | tCKmax high | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 |
| SPD # | tAA (ps) | 16363 | 18750 | 19090 | 20000 | 16666 | 18750 | 19166 | 20000 | 16153 | 18750 | 19230 | 20000 |
| 30 | tAA low | 0xEB | 0x3E | 0x92 | 0x20 | 0x1A | 0x3E | 0xDE | 0x20 | 0x18 | 0x3E | 0x1E | 0x20 |
| 31 | tAA high | 0x3F | 0x49 | 0x4A | 0x4E | 0x41 | 0x49 | 0x4A | 0x4E | 0x3F | 0x49 | 0x4B | 0x4E |
| SPD # | tRCD (ps) | 14545 | 16000 | 16363 | 17500 | 14166 | 16000 | 16666 | 17500 | 14615 | 16000 | 16153 | 17500 |
| 32 | tRCD low | 0xD1 | 0x80 | 0xEB | 0x5C | 0x56 | 0x80 | 0x1A | 0x5C | 0x17 | 0x80 | 0x18 | 0x5C |
| 33 | tRCD high | 0x38 | 0x3E | 0x3F | 0x44 | 0x37 | 0x3E | 0x41 | 0x44 | 0x39 | 0x3E | 0x3F | 0x44 |
| SPD # | tRP (ps) | 14545 | 16000 | 16363 | 17500 | 14166 | 16000 | 16666 | 17500 | 14615 | 16000 | 16153 | 17500 |
| 34 | tRP low | 0xD1 | 0x80 | 0xEB | 0x5C | 0x56 | 0x80 | 0x1A | 0x5C | 0x17 | 0x80 | 0x18 | 0x5C |
| 35 | tRP high | 0x38 | 0x3E | 0x3F | 0x44 | 0x37 | 0x3E | 0x41 | 0x44 | 0x39 | 0x3E | 0x3F | 0x44 |
| SPD # | tRAS (ps) | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 |
| 36 | tRAS low | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |
| 37 | tRAS high | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D |
| SPD # | tRC (ps) | 46545 | 48000 | 48363 | 49500 | 46166 | 48000 | 48666 | 49500 | 46615 | 48000 | 49500 | 49500 |
| 38 | tRC low | 0xD1 | 0x80 | 0xEB | 0x5C | 0x56 | 0x80 | 0x1A | 0x5C | 0x17 | 0x80 | 0x5C | 0x5C |
| 39 | tRC high | 0xB5 | 0xBB | 0xBC | 0xC1 | 0xB4 | 0xBB | 0xBE | 0xC1 | 0xB6 | 0xBB | 0xC1 | 0xC1 |
| SPD # | tWR (ps) | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 |
| 40 | tWR low | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 |
| 41 | tWR high | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 |

## 9.5    Core parameters, DDR5 Multi-Die Packages (3DS) SDRAMs (cont'd)

| | | 5600AN 3DS | 5600B 3DS | 5600BN 3DS | 5600C 3DS | 6000AN 3DS | 6000B 3DS | 6000BN 3DS | 6000C 3DS | 6400AN 3DS | 6400B 3DS | 6400BN 3DS | 6400C 3DS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | tCKmin | 0.357 | 0.357 | 0.357 | 0.357 | 0.333 | 0.333 | 0.333 | 0.333 | 0.312 | 0.312 | 0.312 | 0.312 |
| | tCKmax | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 | 1.010 |
| | tAA | 16.428 | 18.571 | 18.571 | 20.000 | 16.000 | 18.571 | 18.666 | 20.000 | 16.250 | 18.571 | 18.750 | 20.000 |
| | tRCD | 14.285 | 16.000 | 16.428 | 17.500 | 14.000 | 16.000 | 16.000 | 17.500 | 14.375 | 16.000 | 16.250 | 17.500 |
| | tRP | 14.285 | 16.000 | 16.428 | 17.500 | 14.000 | 16.000 | 16.000 | 17.500 | 14.375 | 16.000 | 16.250 | 17.500 |
| | tRAS | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 | 32.000 |
| | tRC | 46.285 | 48.000 | 48.428 | 49.500 | 46.000 | 48.000 | 48.000 | 49.500 | 46.375 | 48.000 | 48.250 | 49.500 |
| | tWR | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 | 30.000 |

| SPD # | tCKmin (ps) | 357 | 357 | 357 | 357 | 333 | 333 | 333 | 333 | 312 | 312 | 312 | 312 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 20 | tCKmin low | 0x65 | 0x65 | 0x65 | 0x65 | 0x4D | 0x4D | 0x4D | 0x4D | 0x38 | 0x38 | 0x38 | 0x38 |
| 21 | tCKmin high | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 | 0x01 |

| SPD # | tCKmax (ps) | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 22 | tCKmax low | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 | 0xF2 |
| 23 | tCKmax high | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 | 0x03 |

| SPD # | tAA (ps) | 16428 | 18571 | 18571 | 20000 | 16000 | 18571 | 18666 | 20000 | 16250 | 18571 | 18750 | 20000 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 30 | tAA low | 0x2C | 0x8B | 0x8B | 0x20 | 0x80 | 0x8B | 0xEA | 0x20 | 0x7A | 0x8B | 0x3E | 0x20 |
| 31 | tAA high | 0x40 | 0x48 | 0x48 | 0x4E | 0x3E | 0x48 | 0x48 | 0x4E | 0x3F | 0x48 | 0x49 | 0x4E |

| SPD # | tRCD (ps) | 14285 | 16000 | 16428 | 17500 | 14000 | 16000 | 16000 | 17500 | 14375 | 16000 | 16250 | 17500 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 32 | tRCD low | 0xCD | 0x80 | 0x2C | 0x5C | 0xB0 | 0x80 | 0x80 | 0x5C | 0x27 | 0x80 | 0x7A | 0x5C |
| 33 | tRCD high | 0x37 | 0x3E | 0x40 | 0x44 | 0x36 | 0x3E | 0x3E | 0x44 | 0x38 | 0x3E | 0x3F | 0x44 |

| SPD # | tRP (ps) | 14285 | 16000 | 16428 | 17500 | 14000 | 16000 | 16000 | 17500 | 14375 | 16000 | 16250 | 17500 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 34 | tRP low | 0xCD | 0x80 | 0x2C | 0x5C | 0xB0 | 0x80 | 0x80 | 0x5C | 0x27 | 0x80 | 0x7A | 0x5C |
| 35 | tRP high | 0x37 | 0x3E | 0x40 | 0x44 | 0x36 | 0x3E | 0x3E | 0x44 | 0x38 | 0x3E | 0x3F | 0x44 |

| SPD # | tRAS (ps) | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 | 32000 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 36 | tRAS low | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |
| 37 | tRAS high | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D | 0x7D |

| SPD # | tRC (ps) | 46285 | 48000 | 48428 | 49500 | 46000 | 48000 | 48000 | 49500 | 46375 | 48000 | 48250 | 49500 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 38 | tRC low | 0xCD | 0x80 | 0x2C | 0x5C | 0xB0 | 0x80 | 0x80 | 0x5C | 0x27 | 0x80 | 0x7A | 0x5C |
| 39 | tRC high | 0xB4 | 0xBB | 0xBD | 0xC1 | 0xB3 | 0xBB | 0xBB | 0xC1 | 0xB5 | 0xBB | 0xBC | 0xC1 |

| SPD # | tWR (ps) | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 40 | tWR low | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 | 0x30 |
| 41 | tWR high | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 | 0x75 |

## 9.6    Parameters with Lower nCK Limits, DDR5 Multi-Die Packages (3DS) SDRAMs

| | | DDR5-3200 3DS | DDR5-3600 3DS | DDR5-4000 3DS | DDR5-4400 3DS | DDR5-4800 3DS | DDR5-5200 3DS | DDR5-5600 3DS | DDR5-6000 3DS | DDR5-6400 3DS |
|---|---|---|---|---|---|---|---|---|---|---|
| tRRD_L_slr (1K) | ps | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 |
| tRRD_L_slr (1K) | nCK | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| tCCD_L_slr | ps | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 | 5000 |
| tCCD_L_slr | nCK | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| tCCD_L_WR_slr | ps | 20000 | 20000 | 20000 | 20000 | 20000 | 20000 | 20000 | 20000 | 20000 |
| tCCD_L_WR_slr | nCK | 32 | 32 | 32 | 32 | 32 | 32 | 32 | 32 | 32 |
| tCCD_L_WR2_slr | ps | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 |
| tCCD_L_WR2_slr | nCK | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| tFAW_slr (1K) | ps | 20000 | 17777 | 16000 | 14545 | 13333 | 12307 | 11428 | 10666 | 10000 |
| tFAW_slr (1K) | nCK | 32 | 32 | 32 | 32 | 32 | 32 | 32 | 32 | 32 |
| tCCD_L_WTR_slr | ps | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 |
| tCCD_L_WTR_slr | nCK | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| tCCD_S_WTR_slr | ps | 2500 | 2500 | 2500 | 2500 | 2500 | 2500 | 2500 | 2500 | 2500 |
| tCCD_S_WTR_slr | nCK | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| tRTP_slr | ps | 7500 | 7500 | 7500 | 7500 | 7500 | 7500 | 7500 | 7500 | 7500 |
| tRTP_slr | nCK | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 |

| SPD# | | | DDR5-3200 3DS | DDR5-3600 3DS | DDR5-4000 3DS | DDR5-4400 3DS | DDR5-4800 3DS | DDR5-5200 3DS | DDR5-5600 3DS | DDR5-6000 3DS | DDR5-6400 3DS |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 71, 70 | tRRD_L_slr (1K) | MSB, LSB | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 |
| 72 | tRRD_L_slr (1K) | nCK | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 |
| 74, 73 | tCCD_L_slr | MSB, LSB | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 | 0x13, 0x88 |
| 75 | tCCD_L_slr | nCK | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 | 0x08 |
| 77, 76 | tCCD_L_WR_slr | MSB, LSB | 0x4E, 0x20 | 0x4E, 0x20 | 0x4E, 0x20 | 0x4E, 0x20 | 0x4E, 0x20 | 0x4E, 0x20 | 0x4E, 0x20 | 0x4E, 0x20 | 0x4E, 0x20 |
| 78 | tCCD_L_WR_slr | nCK | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 |
| 80, 79 | tCCD_L_WR2_slr | MSB, LSB | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 |
| 81 | tCCD_L_WR2_slr | nCK | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 |
| 83, 82 | tFAW_slr (1K) | MSB, LSB | 0x4E, 0x20 | 0x45, 0x71 | 0x3E, 0x80 | 0x38, 0xD1 | 0x34, 0x15 | 0x30, 0x13 | 0x2C, 0xA4 | 0x29, 0xAA | 0x27, 0x10 |
| 84 | tFAW_slr (1K) | nCK | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 | 0x20 |
| 86, 85 | tCCD_L_WTR_slr | MSB, LSB | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 | 0x27, 0x10 |
| 87 | tCCD_L_WTR_slr | nCK | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 | 0x10 |
| 89, 88 | tCCD_S_WTR_slr | MSB, LSB | 0x09, 0xC4 | 0x09, 0xC4 | 0x09, 0xC4 | 0x09, 0xC4 | 0x09, 0xC4 | 0x09, 0xC4 | 0x09, 0xC4 | 0x09, 0xC4 | 0x09, 0xC4 |
| 90 | tCCD_S_WTR_slr | nCK | 0x04 | 0x04 | 0x04 | 0x04 | 0x04 | 0x04 | 0x04 | 0x04 | 0x04 |
| 92, 91 | tRTP_slr | MSB, LSB | 0x1D, 0x4C | 0x1D, 0x4C | 0x1D, 0x4C | 0x1D, 0x4C | 0x1D, 0x4C | 0x1D, 0x4C | 0x1D, 0x4C | 0x1D, 0x4C | 0x1D, 0x4C |
| 93 | tRTP_slr | nCK | 0x0C | 0x0C | 0x0C | 0x0C | 0x0C | 0x0C | 0x0C | 0x0C | 0x0C |

## 9.7    CAS Latency Masks, DDR5 Multi-Die Packages (3DS) SDRAMs

Optional downbins are included. Optional CAS latencies are not included.

| 3DS Speed Bin | SPD[24] | SPD[25] | SPD[26] | SPD[27] | SPD[28] |
|---|---|---|---|---|---|
| 3200AN 3DS | 0x6A | 0x00 | 0x00 | 0x00 | 0x00 |
| 3200B 3DS | 0x62 | 0x00 | 0x00 | 0x00 | 0x00 |
| 3200BN 3DS | 0x62 | 0x00 | 0x00 | 0x00 | 0x00 |
| 3200C 3DS | 0x42 | 0x00 | 0x00 | 0x00 | 0x00 |
| 3600AN 3DS | 0xE2 | 0x01 | 0x00 | 0x00 | 0x00 |
| 3600B 3DS | 0xE2 | 0x01 | 0x00 | 0x00 | 0x00 |
| 3600BN 3DS | 0xC2 | 0x01 | 0x00 | 0x00 | 0x00 |
| 3600C 3DS | 0x42 | 0x01 | 0x00 | 0x00 | 0x00 |
| 4000AN 3DS | 0xEA | 0x07 | 0x00 | 0x00 | 0x00 |
| 4000B 3DS | 0xE2 | 0x07 | 0x00 | 0x00 | 0x00 |
| 4000BN 3DS | 0x42 | 0x07 | 0x00 | 0x00 | 0x00 |
| 4000C 3DS | 0x42 | 0x05 | 0x00 | 0x00 | 0x00 |
| 4400AN 3DS | 0xE2 | 0x1F | 0x00 | 0x00 | 0x00 |
| 4400B 3DS | 0xE2 | 0x1F | 0x00 | 0x00 | 0x00 |
| 4400BN 3DS | 0x42 | 0x1D | 0x00 | 0x00 | 0x00 |
| 4400C 3DS | 0x42 | 0x15 | 0x00 | 0x00 | 0x00 |
| 4800AN 3DS | 0xE2 | 0x7F | 0x00 | 0x00 | 0x00 |
| 4800B 3DS | 0xE2 | 0x7F | 0x00 | 0x00 | 0x00 |
| 4800BN 3DS | 0x42 | 0x75 | 0x00 | 0x00 | 0x00 |
| 4800C 3DS | 0x42 | 0x55 | 0x00 | 0x00 | 0x00 |
| 5200AN 3DS | 0xEA | 0xFF | 0x01 | 0x00 | 0x00 |
| 5200B 3DS | 0xE2 | 0xFF | 0x01 | 0x00 | 0x00 |
| 5200BN 3DS | 0x42 | 0xD5 | 0x01 | 0x00 | 0x00 |
| 5200C 3DS | 0x42 | 0x55 | 0x01 | 0x00 | 0x00 |
| 5600AN 3DS | 0xE2 | 0xFF | 0x05 | 0x00 | 0x00 |
| 5600B 3DS | 0xE2 | 0xFF | 0x05 | 0x00 | 0x00 |
| 5600BN 3DS | 0xC2 | 0x75 | 0x05 | 0x00 | 0x00 |
| 5600C 3DS | 0x42 | 0x55 | 0x05 | 0x00 | 0x00 |
| 6000AN 3DS | 0xEA | 0xFF | 0x15 | 0x00 | 0x00 |
| 6000B 3DS | 0xE2 | 0xFF | 0x15 | 0x00 | 0x00 |
| 6000BN 3DS | 0xE2 | 0xFF | 0x15 | 0x00 | 0x00 |
| 6000C 3DS | 0x42 | 0x55 | 0x15 | 0x00 | 0x00 |
| 6400AN 3DS | 0xEA | 0xFF | 0x55 | 0x00 | 0x00 |
| 6400B 3DS | 0xE2 | 0xFF | 0x5F | 0x00 | 0x00 |
| 6400BN 3DS | 0xE2 | 0x7D | 0x55 | 0x00 | 0x00 |
| 6400C 3DS | 0x42 | 0x55 | 0x55 | 0x00 | 0x00 |

## 9.8    Parameters Defined by Device Density, DDR5 Multi-Die Packages (3DS) SDRAMs

|  |  | SPD[43] | SPD[42] |
| --- | --- | --- | --- |
|  | ns | MSB | LSB |
| tRFC1_slr 3DS, 8 Gb | 195 | 0x00 | 0xC3 |
| tRFC1_slr 3DS, 16 Gb | 295 | 0x01 | 0x27 |
| tRFC1_slr 3DS, 24 Gb | 410 | 0x01 | 0x9A |
| tRFC1_slr 3DS, 32 Gb | 410 | 0x01 | 0x9A |

|  |  | SPD[45] | SPD[44] |
| --- | --- | --- | --- |
|  | ns | MSB | LSB |
| tRFC2_slr 3DS, 8 Gb | 130 | 0x00 | 0x82 |
| tRFC2_slr 3DS, 16 Gb | 160 | 0x00 | 0xA0 |
| tRFC2_slr 3DS, 24 Gb | 220 | 0x00 | 0xDC |
| tRFC2_slr 3DS, 32 Gb | 220 | 0x00 | 0xDC |

|  |  | SPD[47] | SPD[46] |
| --- | --- | --- | --- |
|  | ns | MSB | LSB |
| tRFCsb_slr 3DS, 8 Gb | 115 | 0x00 | 0x73 |
| tRFCsb_slr 3DS, 16 Gb | 130 | 0x00 | 0x82 |
| tRFCsb_slr 3DS, 24 Gb | 190 | 0x00 | 0xBE |
| tRFCsb_slr 3DS, 32 Gb | 190 | 0x00 | 0xBE |

|  |  | SPD[49] | SPD[48] |
| --- | --- | --- | --- |
|  | ns | MSB | LSB |
| tRFC1_dlr 3DS, 8 Gb | 65 | 0x00 | 0x41 |
| tRFC1_dlr 3DS, 16 Gb | 99 | 0x00 | 0x63 |
| tRFC1_dlr 3DS, 24 Gb | 137 | 0x00 | 0x89 |
| tRFC1_dlr 3DS, 32 Gb | 137 | 0x00 | 0x89 |

|  |  | SPD[51] | SPD[50] |
| --- | --- | --- | --- |
|  | ns | MSB | LSB |
| tRFC2_dlr 3DS, 8 Gb | 44 | 0x00 | 0x2C |
| tRFC2_dlr 3DS, 16 Gb | 54 | 0x00 | 0x36 |
| tRFC2_dlr 3DS, 24 Gb | 74 | 0x00 | 0x4A |
| tRFC2_dlr 3DS, 32 Gb | 74 | 0x00 | 0x4A |

|  |  | SPD[53] | SPD[52] |
| --- | --- | --- | --- |
|  | ns | MSB | LSB |
| tRFCsb_dlr 3DS, 8 Gb | 39 | 0x00 | 0x27 |
| tRFCsb_dlr 3DS, 16 Gb | 44 | 0x00 | 0x2C |
| tRFCsb_dlr 3DS, 24 Gb | 64 | 0x00 | 0x40 |
| tRFCsb_dlr 3DS, 32 Gb | 64 | 0x00 | 0x40 |

## 9.9     Blocks 3~6: Module-Specific Section: Bytes 192~447 (0x0C0~0x1BF)

This clause contains SPD bytes which are specific to families DDR5 module families. Module Type Key Byte 3 is used as an index for the encoding of bytes 192~447. The content of bytes 192~447 are described in multiple annexes, one for each memory module family. These bytes are write protected.

## 9.10     Blocks 8~9: Module Supplier's Data: Bytes 512~639 (0x200~0x27F)

These blocks of the SPD contain information specific to the memory module assembly including module manufacturer ID, manufacturing location, and module part number.

## 9.11     Blocks 10~15: End User Programmable, Bytes 640~1023 (0x280~0x3FF)

These blocks of the SPD are not write protected so that end users may program any values into these bytes during system runtime or for system management.

# 10    Annex A.0: Common SPD Bytes for All Module Types (Bytes 192~447, 0x0C0~0x1BF)

This clause describes SPD bytes that are common to all module types. Some bytes, as indicated Table 9, are documented in separate annexes for each module type:

- A.1: Solder Down
- A.2: Unbuffered DIMMs (UDIMM)
- A.3: Registered DIMMs (RDIMM) and Load Reduced DIMMs (LRDIMM)
- A.4: Reserved
- A.5: Differential DIMMs (DDIMM)
- A.6: Non-Volatile DIMMs, -N (NVDIMM-N)
- A.7: Non-Volatile DIMMs, -P (NVDIMM-P)

**Table 9 — Common SPD Bytes for All Module Types**

| SPD Bytes Common to All Module Types | | |
|---|---|---|
| **Byte Number** | | **Function Described** |
| 192 | 0x0C0 | SPD Revision for SPD bytes 192~447 |
| 193 | 0x0C1 | Hashing Sequence |
| 194 | 0x0C2 | SPD Manufacturer ID Code, First Byte |
| 195 | 0x0C3 | SPD Manufacturer ID Code, Second Byte |
| 196 | 0x0C4 | SPD Device Type |
| 197 | 0x0C5 | SPD Device Revision Number |
| 198 | 0x0C6 | PMIC 0 Manufacturer ID Code, First Byte |
| 199 | 0x0C7 | PMIC 0 Manufacturer ID Code, Second Byte |
| 200 | 0x0C8 | PMIC 0 Device Type |
| 201 | 0x0C9 | PMIC 0 Revision Number |
| 202 | 0x0CA | PMIC 1 Manufacturer ID Code, First Byte |
| 203 | 0x0CB | PMIC 1 Manufacturer ID Code, Second Byte |
| 204 | 0x0CC | PMIC 1 Device Type |
| 205 | 0x0CD | PMIC 1 Revision Number |
| 206 | 0x0CE | PMIC 2 Manufacturer ID Code, First Byte |
| 207 | 0x0CF | PMIC 2 Manufacturer ID Code, Second Byte |
| 208 | 0x0D0 | PMIC 2 Device Type |
| 209 | 0x0D1 | PMIC 2 Revision Number |
| 210 | 0x0D2 | Thermal Sensor Manufacturer ID Code, First Byte |
| 211 | 0x0D3 | Thermal Sensor Manufacturer ID Code, Second Byte |
| 212 | 0x0D4 | Thermal Sensor Device Type |
| 213 | 0x0D5 | Thermal Sensor Revision Number |
| 214~229 | 0x0D6~0x0E5 | Reserved |
| 230 | 0x0E6 | Module Nominal Height |
| 231 | 0x0E7 | Module Maximum Thickness |
| 232 | 0x0E8 | Reference Raw Card Used |
| 233 | 0x0E9 | DIMM Attributes |
| 234 | 0x0EA | Module Organization |
| 235 | 0x0EB | Memory Channel Bus Width |
| 236~239 | 0x0EC~0x0EF | Reserved |
| 240~447 | 0x0F0~0x1BF | Module Type Specific Information. These SPD bytes are unique to each module type. See the appropriate annex for the module. |

## 10.1 (Common): SPD Revision for Module Information
Byte 192 (0x0C0)

This byte defines the SPD revision for bytes 192~447. See SPD byte 1 for details.

## 10.2 (Common): Hashing Sequence
Byte 193 (0x0C1)

This byte defines the sequence of device serial numbers used in the hashing of an attribute certificate for the authentication of a memory solution. See JESD316-5 Enhanced Serial Presence Detect Device Specification for details. If authentication is not supported, code as 0x00.

| Hashing Sequence | |
|---|---|
| **Bits 7~3** | **Bits 2~0** |
| **Reserved** | **Serial Number Hashing Sequence** |
| Reserved; must be coded as 00000 | 000 = No authentication<br>001 = Algorithm 1<br>    ESPDs<br>    DRAMs<br>    PMICs<br>    RCDs<br>    TSs<br>    DBs<br>All other codes reserved |

## 10.3 (Common): Module Device Information
Bytes 194~213 (0x0C2~0x0D5)

Table 10 lists the SPD byte addresses for the key components on this assembly.

**Table 10 — SPD Byte Addresses for Common**

| Module Generic Device Information | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Device | Manufacturer ID Code, First Byte | | Manufacturer ID Code, Second Byte | | Device Type | | Device Revision | |
| | Dec | Hex | Dec | Hex | Dec | Hex | Dec | Hex |
| SPD | 194 | 0x0C2 | 195 | 0x0C3 | 196 | 0x0C4 | 197 | 0x0C5 |
| PMIC 0 | 198 | 0x0C6 | 199 | 0x0C7 | 200 | 0x0C8 | 201 | 0x0C9 |
| PMIC 1 | 202 | 0x0CA | 203 | 0x0CB | 204 | 0x0CC | 205 | 0x0CD |
| PMIC 2 | 206 | 0x0CE | 207 | 0x0CF | 208 | 0x0D0 | 209 | 0x0D1 |
| Thermal Sensors (TS) | 210 | 0x0D2 | 211 | 0x0D3 | 212 | 0x0D4 | 213 | 0x0D5 |
| NOTES:<br>SPD = Serial Presence Detect<br>PMIC = Power Management Integrated Circuit | | | | | | | | |

## 10.3 (Common): Module Device Information (Bytes 194~213 (0x0C2~0x0D5) (cont'd)

Manufacturer ID Code: Per JEP106. See SPD bytes 512~513 for an example.

Device types:

| Device Types | | | | |
|---|---|---|---|---|
| | **Bit 7** | **Bit 6** | **Bits 5 ~ 4** | **Bits 3 ~ 0** |
| **Device** | **Devices Installed** | | **Reserved** | **Device Type** |
| SPD | 0 = Not installed<br>1 = Installed | Reserved; must be coded as 0 | Reserved. Must be coded as 00 | 0000: SPD5118 (see JESD300-5)<br>0001: ESPD5216 (see JESD316-5)<br>All other codes reserved |
| PMIC 0 | 0 = Not installed<br>1 = Installed | | | 0000: PMIC5000 (see JESD301-1)<br>0001: PMIC5010 (see JESD301-1)<br>0010: PMIC5100 (see JESD301-2)<br>All other codes reserved |
| PMIC 1 | 0 = Not installed<br>1 = Installed | | | |
| PMIC 2 | 0 = Not installed<br>1 = Installed | | | |
| Thermal Sensors | 0 = TS0 Not installed<br>1 = TS0 Installed | 0 = TS1 Not installed<br>1 = TS1 Installed | | 0000: TS5111 (see JESD302-1)<br>0001: TS5110 (see JESD302-1)<br>All other codes reserved |

Device revision: This byte is coded as a major revision in bits 7~4 and minor revision in bits 3~0, each nibble expressed in BCD. For example, device stepping 3.2 would be coded as 0x32.

## 10.4 (Common): Reserved
## Bytes 214~229 (0x0D6~0x0E5)

## 10.5 (Common): Module Nominal Height
### Byte 230 (0x0E6)

This byte defines the nominal height (A dimension) in millimeters of the fully assembled module including heat spreaders or other added components. Refer to the relevant JEDEC JC-11 module outline (MO) documents for dimension definitions.

| Module Nominal Height | |
|---|---|
| **Bits 7~5** | **Bits 4~0** |
| **Reserved** | **Module Nominal Height max, in mm (baseline height = 15 mm)** |
| Reserved; must be coded as 000 | 00000 = height ≤ 15 mm<br>00001 = 15 < height ≤ 16 mm<br>00010 = 16 < height ≤ 17 mm<br>00011 = 17 < height ≤ 18 mm<br>00100 = 18 < height ≤ 19 mm<br>...<br>01010 = 24 < height ≤ 25 mm<br>01011 = 25 < height ≤ 26 mm<br>...<br>01111 = 29 < height ≤ 30 mm<br>10000 = 30 < height ≤ 31 mm<br>10001 = 31 < height ≤ 32 mm<br>...<br>11111 = 45 mm < height |

**Examples:**

| Nominal Module Height | Coding, bits 4 ~ 0 | Meaning |
|---|---|---|
| **mm** | **Binary** | **mm** |
| 18.75 | 00100 | 18 < height ≤ 19 mm |
| 25.40 | 01011 | 25 < height ≤ 26 mm |
| 30.00 | 01111 | 29 < height ≤ 30 mm |
| 30.25 | 10000 | 30 < height ≤ 31 mm |
| 31.25 | 10001 | 31 < height ≤ 32 mm |

## 10.6    (Common): Module Maximum Thickness
##          Byte 231 (0x0E7)

This byte defines the maximum thickness in millimeters of the fully assembled module including heat spreaders or other added components above the module circuit board surface, rounding up to the next integer. Refer to the relevant JEDEC JC-11 module outline (MO) documents for dimension definitions.

| Module Maximum Thickness | |
| --- | --- |
| **Bits 7 ~ 4** | **Bits 3 ~ 0** |
| **Module Maximum Thickness max, Back, in mm (baseline thickness = 1 mm)** | **Module Maximum Thickness max, Front, in mm (baseline thickness = 1 mm)** |
| 0000 = thickness ≤ 1 mm<br>0001 = 1 < thickness ≤ 2 mm<br>0010 = 2 < thickness ≤ 3 mm<br>0011 = 3 < thickness ≤ 4 mm<br>...<br>1110 = 14 < thickness ≤ 15 mm<br>1111 = 15 < thickness | 0000 = thickness ≤ 1 mm<br>0001 = 1 < thickness ≤ 2 mm<br>0010 = 2 < thickness ≤ 3 mm<br>0011 = 3 < thickness ≤ 4 mm<br>...<br>1110 = 14 < thickness ≤ 15 mm<br>1111 = 15 < thickness |

## 10.7    (Common): Reference Raw Card Used
## Byte 232 (0x0E8)

This byte indicates which JEDEC reference design raw card was used as the basis for the module assembly, if any. Special reference raw card indicator, ZZ, is used when no JEDEC standard raw card reference design was used as the basis for the module design. Pre-production modules should be encoded as revision 0 in bits 7~5.

| Reference Raw Card Used | |
| --- | --- |
| **Bit 7~5** | **Bits 4~0** |
| **Design Revision** | **Reference Design** |
| 000 = Revision 0<br>001 = Revision 1<br>010 = Revision 2<br>011 = Revision 3<br>100 = Revision 4<br>101 = Revision 5<br>110 = Revision 6<br><br>111 = Reserved | 00000 = A<br>00001 = B<br>00010 = C<br>00011 = D<br>00100 = E<br>00101 = F<br>00110 = G<br>00111 = H<br>01000 = J<br>01001 = K<br>01010 = L<br>01011 = M<br>01100 = N<br>01101 = P<br>01110 = R<br>01111 = T<br>10000 = U<br>10001 = V<br>10010 = W<br>10011 = Y<br>10100 = AA<br>10101 = AB<br>10110 = AC<br>10111 = AD<br>11000 = AE<br>11001 = AF<br>11010 = AG<br>11011 = AH<br>11100 = AJ<br>11101 = AK<br><br>11110 = Reserved<br><br>11111 = ZZ (No reference used) |
| 11111111 = ZZ (no JEDEC reference raw card design used) | |

## 10.8 (Common): DIMM Attributes
## Byte 233 (0x0E9)

This byte indicates the number of rows of DRAM packages (monolithic or 3D stacked) parallel to edge connector (independent of DRAM orientation) on each side of the printed circuit board, and whether the assembly is covered in a heat spreader. The temperature range applies to the SDRAM components on the module. All other support components on the module must remain within their respective operating temperature ranges when the case temperature of the SDRAMs are at the minimum and maximum values.

| DIMM Attributes | | | |
|---|---|---|---|
| Bits 7~4 | Bit 3 | Bit 2 | Bits 1~0 |
| Operating Temperature Range[1] | Reserved | Heat Spreader | # of rows of DRAMs on Module |
| 0000 = A1T (-40 to +125 °C)<br>0001 = A2T (-40 to +105 °C)<br>0010 = A3T (-40 to +85 °C)<br>0011 = IT (-40 to +95 °C)<br>0100 = ST (-25 to +85 °C)<br>0101 = ET (-25 to +105 °C)<br>0110 = RT (0 to +45 °C)<br>0111 = NT (0 to +85 °C)<br>1000 = XT (0 to +95 °C)[2]<br>All other codings reserved | Reserved; must be coded as 0 | 0 = Not installed<br>1 = Installed | 00 = Undefined<br>01 = 1 row<br>10 = 2 rows<br>11 = Reserved |
| NOTE 1    See JESD402-1 for details<br>NOTE 2    Typical DDR5 SDRAM thermal specification is XT, 0 to +95 °C, code 1000 in bits 7~4 | | | |

## 10.9 (Common): Module Organization
## Byte 234 (0x0EA)

This byte describes the organization of the SDRAM module. Bits 5~3 encode the number of package ranks on the module. Bit 6 describes whether the assembly has the same SDRAM density on all ranks or has different SDRAM densities on even and odd ranks.

| Module Organization | | | |
|---|---|---|---|
| Bit 7 | Bit 6 | Bits 5~3 | Bits 2~0 |
| Reserved | Rank Mix | Number of Package Ranks per Channel | Reserved |
| Reserved; must be coded as 0 | 0 = Symmetrical<br>1 = Asymmetrical | Bit [5, 4, 3] :<br>    000 = 1 Package Rank<br>    001 = 2 Package Ranks<br>    010 = 3 Package Ranks<br>    011 = 4 Package Ranks<br>    100 = 5 Package Ranks<br>    101 = 6 Package Ranks<br>    110 = 7 Package Ranks<br>    111 = 8 Package Ranks | Reserved; must be coded as 000 |

## 10.9    (Common): Module Organization (Byte 234 (0x0EA) (cont'd)

"Package ranks per DIMM" refers to the collections of devices on the module sharing common chip select signals (across the data width of the DIMM channel), either from the edge connector for unbuffered modules or from the outputs of a registering clock driver for RDIMMs and LRDIMMs.

"Logical rank" refers the individually addressable die in a 3DS stack and has no meaning for monolithic SDRAMs, however, for purposes of calculating the capacity of the module, one should treat monolithic SDRAMs as having one logical rank per package rank.

Byte 242 Bit 6 = 0 defines a memory module as being "symmetrical" where all SDRAM devices on the module are "First SDRAM" type per SPD bytes 4~7.

Examples shown cover 1 and 2 rank modules, i.e., codes 000 or 001 in SPD byte 234 bits 5~3; for higher package rank counts, apply the same logic.
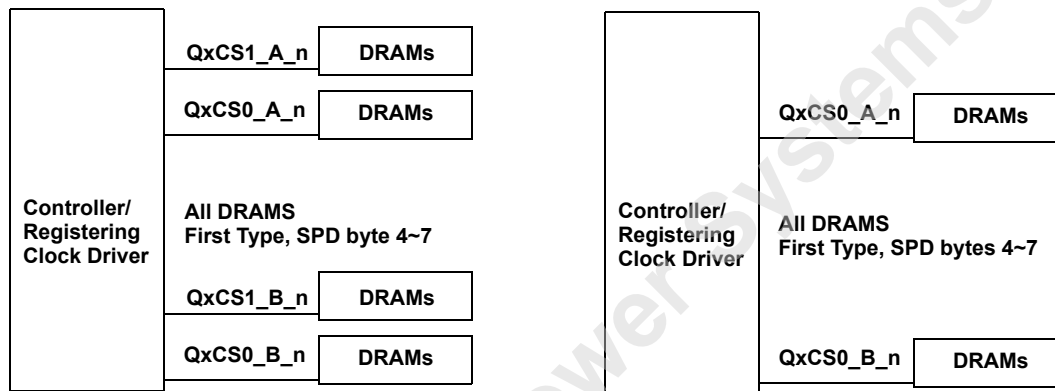


**Symmetrical Configurations (byte 234 Bit 6 = 0)**

| Rank Matrix for Symmetrical Modules (Byte 234 Bit 6 = 0) | | |
|---|---|---|
| SDRAM Package Type | # Package Ranks per Channel | # Logical Ranks per Channel |
| SPD Byte 4 Bits 7~5 | SPD Byte 234 Bits 5~3 | Calculated |
| SDP | 1 | 1 |
|  | 2 | 2 |
| 2H 3DS | 1 | 2 |
|  | 2 | 4 |
| 4H 3DS | 1 | 4 |
|  | 2 | 8 |
| 8H 3DS | 1 | 8 |
|  | 2 | 16 |
| 16H 3DS | 1 | 16 |
|  | 2 | 32 |
| Logical Ranks per Channel = # Package Ranks per Channel * # Logical Ranks per Package Rank | | |
| Bytes 8~10 coded as 0x00 for all symmetrical configurations | | |

## 10.9    (Common): Module Organization (Byte 234 (0x0EA) (cont'd)

Byte 234 Bit 6 = 1 defines a memory module as being "asymmetrical", where all DRAMs in even package ranks (0, 2, 4, 6) are of "First SDRAM" type per SPD bytes 4~7, and all DRAMs in odd package ranks (1, 3, 5, 7) are of "Second SDRAM" type as described in SPD bytes 8~11.

The example shown covers the case of a 2 rank module, i.e., code 001 in SPD byte 234 bits 5~3; for higher package rank counts, apply the same logic for even (Primary) and odd (Secondary) ranks.

The I/O width of all ranks (SPD byte 6 bits 7~5 and SPD byte 10 bits 7~5) in an asymmetrical assembly must be the same, e.g., all x4 or all x8.
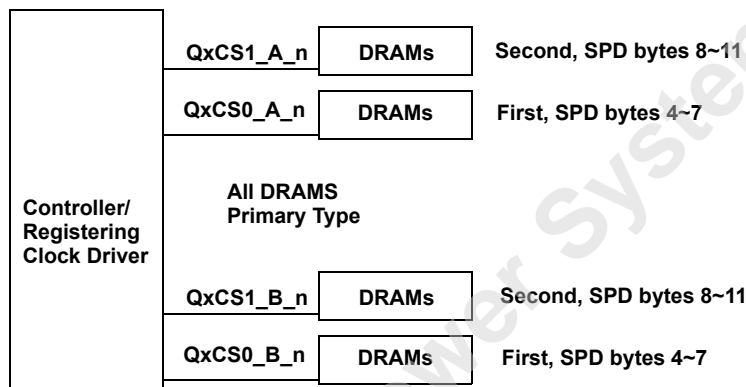


**Asymmetrical Configurations (byte 234 Bit 6 = 1)**

| Rank Mix for Asymmetrical Modules (Byte 242 Bit 6 = 1) | | | |
|---|---|---|---|
| First SDRAM Package Type | Second SDRAM Package Type | # Package Ranks per Channel | # Logical Ranks per Channel |
| SPD Byte 4 Bits 7~5 | SPD Byte 8 Bits 7~5 | SPD Byte 234 Bits 5~3 | Calculated |
| SDP | 2H 3DS | 2 | 3 |
| 2H 3DS | SDP | 2 | 3 |
| SDP | 4H 3DS | 2 | 5 |
| 4H 3DS | SDP | 2 | 5 |
| ... | ... | ... | ... |
| 8H 3DS | 16H 3DS | 2 | 24 |
| 16H 3DS | 8H 3DS | 2 | 24 |
| Logical Ranks per DIMM = # Logical Ranks in Primary SDRAM type + # Logical Ranks in Secondary SDRAM type | | | |

### 10.10    (Common): Memory Channel Bus Width
####       Byte 235 (0x0EB)

This byte describes the width of the SDRAM memory bus on the module. Bits 2~0 encode the primary bus width. Bits 4~3 encode the bus extensions such as parity or ECC. Bits 6~5 defines the number of channels on each module.

| Memory Channel Bus Width | | | |
|---|---|---|---|
| **Bit 7** | **Bits 6~5** | **Bits 4~3** | **Bits 2~0** |
| **Reserved** | **Number of Channels per DIMM** | **Bus width extension per Channel, in bits** | **Primary bus width per Channel, in bits** |
| Reserved; must be coded as 0 | Bit [6, 5] :<br>  00 = 1 channel<br>  01 = 2 channels<br>  All others reserved | Bit [4, 3] :<br>  00 = 0 bits (no extension)<br>  01 = 4 bits<br>  10 = 8 bits<br>  All others reserved | Bit [2, 1, 0] :<br>  000 = 8 bits<br>  001 = 16 bits<br>  010 = 32 bits<br>  011 = 64 bits<br>  All others reserved |

**Examples:**

| Number of Channels Per DIMM | Primary Bus Width per Channel | Bus Width Extension Per Bus | Byte 235 Coding | Total Data Width per DIMM | Standard Usage |
|---|---|---|---|---|---|
| 2 | 32 | 0 | 0 01 00 010 | 64 | SODIMM, UDIMM |
| 2 | 32 | 4 | 0 01 01 010 | 72 | ECC SODIMM, ECC UDIMM |
| 2 | 32 | 8 | 0 01 10 010 | 80 | EC8 RDIMM, LRDIMM, NVDIMM |
| 2 | 32 | 4 | 0 01 01 010 | 72 | EC4 RDIMM |

### 10.10.1    Calculating Module DRAM Capacity

The total memory capacity of the DRAM on a DIMM may be calculated from SPD values. Table 11 details the SPD bytes needed for the calculation.

**Table 11 — SPD Bytes Needed for Total Memory Capacity Calculation**

| Symmetry | Package Ranks per Channel | SDRAM Density per Die | Die per Package | SDRAM I/O Width | Primary Bus Width per Channel | Number of Channels per DIMM |
|---|---|---|---|---|---|---|
| 0 = Symmetrical | All | SPD byte 4 Bits 4~0 | SPD byte 4 Bits 7~5 | SPD byte 6 Bits 7~5 | SPD byte 235 Bits 2~0 | SPD byte 235 Bits 6~5 |
| 1 = Asymmetrical | Even (0, 2, 4, 6) | SPD byte 4 Bits 4~0 | SPD byte 4 Bits 7~5 | SPD byte 6 Bits 7~5 | SPD byte 235 Bits 2~0 | SPD byte 235 Bits 6~5 |
| | Odd (1, 3, 5, 7) | SPD byte 8 Bits 4~0 | SPD byte 8 Bits 7~5 | SPD byte 10 Bits 7~5 | SPD byte 235 Bits 2~0 | SPD byte 235 Bits 6~5 |

### 10.10.1 Calculating Module DRAM Capacity (cont'd)

To calculate the total capacity in bytes for a symmetric module, the following math applies:

Capacity in bytes =

Number of channels per DIMM *

Primary bus width per channel / SDRAM I/O Width *

Die per package *

SDRAM density per die / 8 *

Package ranks per channel

To calculate the total capacity in bytes for an asymmetric module, the following math applies:

Capacity in bytes =

Capacity of even ranks (first SDRAM type) +

Capacity of odd ranks (second SDRAM type)

Commonly, parity or ECC are not counted in total module capacity, though they can also be included by adding the bus width extension in SPD byte 235 bits 4~3 to the primary bus width in the previous examples.

### 10.11 (Common): Reserved
### Bytes 236~239 (0x0EC~0EF)

Reserved - must be coded as 0x00

### 10.12 (Common): Module Type Specific Information
### Bytes 240~447 (0x0F0~0x1BF)

These bytes are specific to each module type. See the appropriate annex for definition.

# 11 Annex A.1: Module Specific Bytes for Solder Down (Bytes 192~447, 0x0C0~0x1BF)

## 11.1 Revision 1.0: Byte 192 Coded as 0x10

This clause defines the encoding of SPD bytes 192~447 when Memory Technology Key Byte 2 contains the value 0x12 and Module Type Key Byte 3 contains any of the following:

- 0xHB, Solder Down
- where H refers to the hybrid memory architecture, if any present

See Annex A.0 for common SPD bytes 192~239 (0x0C0~0EF).

| Module Specific SPD Bytes for Solder Down Memory | | |
|---|---|---|
| Byte Number | | Function Described |
| 240~447 | 0x0F0~0x1BF | Reserved |

## 11.2 Bytes 240~447 (0x0F8~0x1BF) (Solder Down):

Reserved - must be coded as 0x00.

# 12 Annex A.2: Module Specific Bytes for Unbuffered Memory Module Types (Bytes 192~447, 0x0C0~0x1BF)

## 12.1 Revision 1.0: Byte 192 Coded as 0x10

This clause defines the encoding of SPD bytes 192~447 when Memory Technology Key Byte 2 contains the value 0x12 and Module Type Key Byte 3 contains any of the following:

- 0x02, UDIMM
- 0x03, SODIMM

**See Annex A.0 for common SPD bytes 192~239 (0x0C0~0EF).**

| Module Specific SPD Bytes for Unbuffered Module Types | | |
|---|---|---|
| Byte Number | | Function Described |
| 240 | 0x0F0 | Clock Driver Manufacturer ID Code, First Byte |
| 241 | 0x0F1 | Clock Driver Manufacturer ID Code, Second Byte |
| 242 | 0x0F2 | Clock Driver Device Type |
| 243 | 0x0F3 | Clock Driver Revision Number |
| 244~445 | 0x0F4~0x1BD | Reserved |

## 12.2 (Unbuffered): Module Specific Device Information Bytes 240~243 (0x0F0~0x0F3)

This table lists the SPD byte addresses for the key components on this assembly.

**Table 12 — SPD Byte Addresses for Unbuffered**

| Module Specific Device Information | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Device | Manufacturer ID Code, First Byte | | Manufacturer ID Code, Second Byte | | Device Type | | Device Revision | |
| | Dec | Hex | Dec | Hex | Dec | Hex | Dec | Hex |
| Clock Driver (CK) | 240 | 0x0F0 | 241 | 0x0F1 | 242 | 0x0F2 | 243 | 0x0F3 |

## 12.2    (Unbuffered): Module Specific Device Information (cont'd)
### Bytes 240~243 (0x0F0~0x0F3)

Manufacturer ID Code: Per JEP106. See SPD bytes 512~513 for an example.
Device types:

| Module Specific Device Types | | | | |
|---|---|---|---|---|
| | **Bit 7** | **Bit 6** | **Bits 5 ~ 4** | **Bits 3 ~ 0** |
| **Device** | **Devices Installed** | | **Reserved** | **Device Type** |
| Clock Driver (CK) | 0 = Not installed<br>1 = Installed | Reserved; must be coded as 0 | Reserved. Must be coded as 00 | 0000: DDR5CK01 (see JESD82-531)<br>All other codes reserved |

Device revision: This byte is coded as a major revision in bits 7~4 and minor revision in bits 3~0, each nibble expressed in BCD. For example, device stepping 3.2 would be coded as 0x32.

## 12.3    (Unbuffered): Reserved
### Bytes 244~447 (0x0F4~0x1BF)

Reserved - must be coded as 0x00.

# 13    Annex A.3: Module Specific Bytes for Registered (RDIMM) and Load Reduced (LRDIMM) Memory Module Types

## 13.1    (Bytes 192~447, 0x0C0~0x1BF)
## Revision 1.0: Byte 192 Coded as 0x10

This clause defines the encoding of SPD bytes 192~447 when Memory Technology Key Byte 2 contains the value 0x12 and Module Type Key Byte 3 contains any of the following:

- 0xH1, RDIMM
- 0xH4, LRDIMM
- where H refers to the hybrid memory architecture, if any present on the module.

Table entries highlighted in light blue are LRDIMM specific.

The ECC data width for RDIMMs may vary from design to design. Some RDIMMs support 8-bit ECC per channel (EC8: 2 channels of 40 bits wide each) while other RDIMMs support 4-bit ECC per channel (EC4: 2 channels of 36 bits wide each). To determine which configuration of ECC is supported by this module design, look to the following SPD bytes:

- SPD byte 232: Reference Raw Card Used
- SPD byte 235: Memory Channel Bus Width, especially bits 4~3, bus width extension

RDIMMs and LRDIMMs may support symmetrical or asymmetrical memory ranks. Symmetrical means that all ranks on the DIMM are identical. Asymmetrical means that even ranks and odd ranks may contain SDRAMs of different per-device capacity. This rule applies across both A and B channels; all devices on both A and B channels within any rank must be identical. See Annex A.0 Common SPD Bytes for details. In particular, symmetry is defined by the following SPD byte:

- SPD byte 234: Module Organization, especially bit 6, Rank Mix

 All note that other SDRAM characteristics may vary between even and odd ranks in asymmetrical designs, such as Refresh Management (RFM) settings.

## 13.1 (Bytes 192~447, 0x0C0~0x1BF), Revision 1.0: Byte 192 Coded as 0x10 (cont'd)

**See Annex A.0 for common SPD bytes 192~239 (0x0C0~0EF).**

| Module Specific SPD Bytes for Registered and Load Reduced Module Types | | |
|---|---|---|
| Byte Number | | Function Described |
| 240 | 0x0F0 | Registering Clock Driver Manufacturer ID Code, First Byte |
| 241 | 0x0F1 | Registering Clock Driver Manufacturer ID Code, Second Byte |
| 242 | 0x0F2 | Register Device Type |
| 243 | 0x0F3 | Register Revision Number |
| 244 | 0x0F4 | Data Buffer Manufacturer ID Code, First Byte |
| 245 | 0x0F5 | Data Buffer Manufacturer ID Code, Second Byte |
| 246 | 0x0F6 | Data Buffer Device Type |
| 247 | 0x0F7 | Data Buffer Revision Number |
| 248 | 0x0F8 | RCD-RW08 Clock Driver Enable |
| 249 | 0x0F9 | RCD-RW09 Output Address and Control Enable |
| 250 | 0x0FA | RCD-RW0A QCK Driver Characteristics |
| 251 | 0x0FB | RCD-RW0B |
| 252 | 0x0FC | RCD-RW0C QxCA and QxCS_n Driver Characteristics |
| 253 | 0x0FD | RCD-RW0D Data Buffer Interface Driver Characteristics |
| 254 | 0x0FE | RCD-RW0E QCK, QCA and QCS Output Slew Rate |
| 255 | 0x0FF | RCD-RW0F BCK, BCOM, and BCS Output Slew Rate |
| 256 | 0x100 | DB-RW86 DQS RTT Park Termination |
| 257~447 | 0x101~0x1BF | Reserved |

## 13.2 (RDIMM/LRDIMM): Module Specific Device Information Bytes 240~247 (0x0F0~0x0F7)

This table lists the SPD byte addresses for the key components on this assembly.

**Table 13 — SPD Byte Addresses for RDIMM / LRDIMM**

| Module Specific Device Information | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Device | Manufacturer ID Code, First Byte | | Manufacturer ID Code, Second Byte | | Device Type | | Device Revision | |
| | Dec | Hex | Dec | Hex | Dec | Hex | Dec | Hex |
| Registering Clock Driver (RCD) | 240 | 0x0F0 | 241 | 0x0F1 | 242 | 0x0F2 | 243 | 0x0F3 |
| Data Buffers (DB) | 244 | 0x0F4 | 245 | 0x0F5 | 246 | 0x0F6 | 247 | 0x0F7 |

## 13.2 (RDIMM/LRDIMM): Module Specific Device Information (cont'd)
Bytes 240~247 (0x0F0~0x0F7)

Manufacturer ID Code: Per JEP106. See SPD bytes 512~513 for an example.
Device types:

| Module Specific Device Types | | | | |
|---|---|---|---|---|
| | **Bit 7** | **Bit 6** | **Bits 5 ~ 4** | **Bits 3 ~ 0** |
| **Device** | **Devices Installed** | | **Reserved** | **Device Type** |
| Registering Clock Driver (RCD) | 0 = Not installed<br>1 = Installed | Reserved; must be coded as 0 | Reserved. Must be coded as 00 | 0000: DDR5RCD01 (see JESD82-511)<br>0001: DDR5RCD02 (see JESD82-512)<br>All other codes reserved |
| Data Buffers (DB) | 0 = Not installed (RDIMM)<br>1 = Installed (LRDIMM) | | | 0000: DDR5DB01 (see JESD82-521)<br>0001: DDR5DB02 (see JESD82-522)<br>All other codes reserved<br>RDIMM: code as 0000 |

Device revision: This byte is coded as a major revision in bits 7~4 and minor revision in bits 3~0, each nibble expressed in BCD. For example, device stepping 3.2 would be coded as 0x32.

## 13.3 (RDIMM/LRDIMM): RCD-RW08 Clock Driver Enable
Byte 248 (0x0F8)

Byte defined as per JESD82-511 DDR5RCD01 standard.

| RCD-RW08: Clock Driver Enable | | | | | | |
|---|---|---|---|---|---|---|
| **Bits 7~6** | **Bit 5** | **Bit 4** | **Bit 3** | **Bit 2** | **Bit 1** | **Bit 0** |
| **Reserved** | **BCK_t/ BCK_c** | **Reserved** | **QDCK_t/ QDCK_c** | **QCCK_t/ QCCK_c** | **QBCK_t/ QBCK_c** | **QACK_t/ QACK_c** |
| Reserved; must be coded as 00 | 0 = Enabled<br>1 = Disabled | Reserved; must be coded as 0 | 0 = Enabled<br>1 = Disabled | 0 = Enabled<br>1 = Disabled | 0 = Enabled<br>1 = Disabled | 0 = Enabled<br>1 = Disabled |

## 13.4 (RDIMM/LRDIMM): RCD-RW09 Output Address and Control Enable Byte 249 (0x0F9)

Byte defined as per JESD82-511 DDR5RCD01 standard.

| RCD-RW09: Output Address and Control Enables | | | |
|---|---|---|---|
| **Bit 7** | **Bit 6** | **Bit 5** | **Bit 4** |
| Reserved | QBCS[1:0]_n output | QACS[1:0]_n output | Q[B:A]CA13 output driver |
| Reserved; must be coded as 0 | 0 = Enabled<br>1 = Disabled | 0 = Enabled<br>1 = Disabled | 0 = Enabled<br>1 = Disabled |
| **Bit 3** | **Bit 2** | **Bit 1** | **Bit 0** |
| BCS_n, BCOM[2:0] and BRST_n outputs | DCS1_n input buffer and QxCS1_n outputs | QBCA outputs | QACA outputs |
| 0 = Enabled (LRDIMM)<br>1 = Disabled (RDIMM) | 0 = Enabled<br>1 = Disabled | 0 = Enabled<br>1 = Disabled | 0 = Enabled<br>1 = Disabled |

## 13.5 (RDIMM/LRDIMM): RCD-RW0A QCK Driver Characteristics Byte 250 (0x0FA)

Byte defined as per JESD82-511 DDR5RCD01 standard.

| RCD-RW0A: QCK Driver Characteristics | | | |
|---|---|---|---|
| **Bits 7~6** | **Bits 5~4** | **Bits 3~2** | **Bits 1~0** |
| QDCK_t/QDCK_c | QCCK_t/QCCK_c | QBCK_t/QBCK_c | QACK_t/QACK_c |
| 00 = RZQ/12 (20Ω)<br>01 = RZQ/17 (14Ω)<br>10 = RZQ/24 (10Ω)<br>11 = Reserved | 00 = RZQ/12 (20Ω)<br>01 = RZQ/17 (14Ω)<br>10 = RZQ/24 (10Ω)<br>11 = Reserved | 00 = RZQ/12 (20Ω)<br>01 = RZQ/17 (14Ω)<br>10 = RZQ/24 (10Ω)<br>11 = Reserved | 00 = RZQ/12 (20Ω)<br>01 = RZQ/17 (14Ω)<br>10 = RZQ/24 (10Ω)<br>11 = Reserved |

## 13.6 (RDIMM/LRDIMM): RCD-RW0B
### Byte 251 (0x0FB)

| RCD-RW0B: |
|---|
| **Bits 7~0** |
| **Reserved** |
| Reserved. Must be coded as 0x00. |

## 13.7 (RDIMM/LRDIMM): RCD-RW0C QxCA and QxCS_n Driver Characteristics Byte 252 (0x0FC)

Byte defined as per JESD82-511 DDR5RCD01 standard.

| RCD-RW0C: QxCA and QxCS_n Driver Characteristics | | | |
|---|---|---|---|
| **Bits 7~6** | **Bits 5~4** | **Bits 3~2** | **Bits 1~0** |
| **Reserved** | **Driver Strength QxCS0_n, QxCS1_n** | **Reserved** | **Driver Strength Address/Command for both A and B outputs** |
| Reserved. Must be coded as 00 | 00 = RZQ/12 (20 Ω)<br>01 = RZQ/17 (14 Ω)<br>10 = RZQ/24 (10 Ω)<br>11 = Reserved | Reserved. Must be coded as 00 | 00 = RZQ/12 (20 Ω)<br>01 = RZQ/17 (14 Ω)<br>10 = RZQ/24 (10 Ω)<br>11 = Reserved |

## 13.8 (RDIMM/LRDIMM): RCD-RW0D Data Buffer Interface Driver Characteristics Byte 253 (0x0FD)

Byte defined as per JESD82-511 DDR5RCD01 standard. These values are LRDIMM specific; RDIMMs shall encode as 0x00.

| RCD-RW0D: Data Buffer Interface Driver Characteristics | | | |
|---|---|---|---|
| **Bits 7~5** | **Bits 4~3** | **Bit 2** | **Bits 1~0** |
| **Reserved** | **Driver Strength BCK_t/BCK_c** | **Reserved** | **Driver Strength BCOM[2:0], BCS_n** |
| Reserved. Must be coded as 000 | 00 = RZQ/12 (20 Ω)<br>01 = RZQ/17 (14 Ω)<br>10 = RZQ/24 (10 Ω)<br>11 = Reserved | Reserved. Must be coded as 0 | 00 = RZQ/12 (20 Ω)<br>01 = RZQ/17 (14 Ω)<br>10 = Reserved)<br>11 = Reserved |

## 13.9 (RDIMM/LRDIMM): RCD-RW0E QCK, QCA, and QCS Output Slew Rate Byte 254 (0x0FE)

Byte defined as per JESD82-511 DDR5RCD01 standard.

| RCD-RW0E: QCK, QCA, and QCS Output Slew Rate | | | |
|---|---|---|---|
| **Bits 7~6** | **Bits 5~4** | **Bits 3~2** | **Bits 1~0** |
| **Reserved** | **Q[B:A]CS[1:0]_n Single Ended Slew Rate** | **Q[B:A]CA[13:0] Single Ended Slew Rate** | **QCK[D:A]_t/QCK[D:A]_c Differential Slew Rate** |
| Reserved. Must be coded as 00 | 00 = Moderate (4-7 V/ns)<br>01 = Fast (6-10 V/ns)<br>10 = Slow (2.7-4.5 V/ns)<br>11 = Reserved | 00 = Moderate (4-7 V/ns)<br>01 = Fast (6-10 V/ns)<br>10 = Slow (2.7-4.5 V/ns)<br>11 = Reserved | 00 = Moderate (12-20 V/ns)<br>01 = Fast (14-27 V/ns)<br>10 = Reserved<br>11 = Reserved |

## 13.10 (RDIMM/LRDIMM): RCD-RW0F BCK, BCOM, and BCS Output Slew Rate Byte 255 (0x0FF)

Byte defined as per JESD82-511 DDR5RCD01 standard. These values are LRDIMM specific; RDIMMs shall encode as 0x00.

| RCD-RW0F: BCK, BCOM, and BCS Output Slew Rate | | |
|---|---|---|
| Bits 7~4 | Bits 3~2 | Bits 1~0 |
| Reserved | BCK_t/BCK_c<br>Differential Slew Rate | BCOM[2:0], BCS_n<br>Single Ended slew rate |
| Reserved. Must be coded as 0000 | 00 = Moderate (12-20 V/ns)<br>01 = Fast (14-27 V/ns)<br>10 = Reserved<br>11 = Reserved | 00 = Moderate (3.7-6.7 V/ns)<br>01 = Fast (5.2-9.2 V/ns)<br>10 = Slow (2.5-4.3 V/ns)<br>11 = Reserved |

## 13.11 (RDIMM/LRDIMM): DB-RW86 DQS RTT Park Termination Byte 256 (0x100)

Byte defined as per JESD82-521 DDR5DB01 standard. These values are LRDIMM specific; RDIMMs shall encode as 0x00.

| DB-RW86: DQS RTT Park Termination | |
|---|---|
| Bits 7~3 | Bits 2~0 |
| Reserved | DQS RTT Park Termination |
| Reserved. Must be coded as 00000 | 000 = RTT_OFF (default)<br>001 = RZQ/1 (240 Ω)<br>010 = RZQ/2 (120 Ω)<br>011 = RZQ/3 (80 Ω)<br>100 = RZQ/4 (60 Ω)<br>101 = RZQ/5 (48 Ω)<br>110 = RZQ/6 (40 Ω)<br>111 = RZQ/7 (34 Ω) |

## 13.12 (RDIMM/LRDIMM): Reserved Bytes 257~447 (0x101~0x1BF)

Reserved - must be coded as 0x00.

## 14 Annex A.4: Reserved (Bytes 192~447, 0x0C0~0x1BF)

### 14.1 (Reserved): Reserved
Bytes 192~447 (0x0C0~1BF)

Reserved - must be coded as 0x00.

# 15 Annex A.5: Module Specific Bytes for Differential Memory Module Types (Bytes 192~447, 0x0C0~0x1BF)

## 15.1 Revision 1.0: Byte 192 Coded as 0x10

This clause defines the encoding of SPD bytes 192~447 when Memory Technology Key Byte 2 contains the value 0x0C and Module Type Key Byte 3 contains any of the following:

- 0xHA, DDIMM
- where H refers to the hybrid memory architecture, if any present on the module

**See Annex A.0 for common SPD bytes 192~239 (0x0C0~0EF).**

| Module Specific SPD Bytes for Differential Module Types | | |
|---|---|---|
| **Byte Number** | | **Function Described** |
| 240 | 0x0F0 | Differential Memory Buffer Manufacturer ID Code, First Byte |
| 241 | 0x0F1 | Differential Memory Buffer Manufacturer ID Code, Second Byte |
| 242 | 0x0F2 | Differential Memory Buffer Device Type |
| 243 | 0x0F3 | Differential Memory Buffer Revision Number |
| 246~445 | 0x0F6~0x1BD | Reserved |

## 15.2 (Registered): Module Specific Device Information Bytes 240~243 (0x0F0~0x0F3)

This table lists the SPD byte addresses for the key components on this assembly.

**Table 14 — SPD Byte Addresses for Registered**

| Module Specific Device Information | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Device** | **Manufacturer ID Code, First Byte** | | **Manufacturer ID Code, Second Byte** | | **Device Type** | | **Device Revision** | |
| | Dec | Hex | Dec | Hex | Dec | Hex | Dec | Hex |
| Differential Memory Buffer (DMB) | 240 | 0x0F0 | 241 | 0x0F1 | 242 | 0x0F2 | 243 | 0x0F3 |

## 15.2 (Registered): Module Specific Device Information, Bytes 240~243 (0x0F0~0x0F3) (cont'd)

Manufacturer ID Code: Per JEP106. See SPD bytes 512~513 for an example.

Device types:

| Module Specific Device Types | | | | |
|---|---|---|---|---|
| | **Bit 7** | **Bit 6** | **Bits 5 ~ 4** | **Bits 3 ~ 0** |
| **Device** | **Devices Installed** | | **Reserved** | **Device Type** |
| Differential Memory Buffer (DMB) | 0 = Not installed 1 = Installed | Reserved; must be coded as 0 | Reserved. Must be coded as 00 | 0000: DMB501 (see JESD303-1) All other codes reserved |

Device revision: This byte is coded as a major revision in bits 7~4 and minor revision in bits 3~0, each nibble expressed in BCD. For example, device stepping 3.2 would be coded as 0x32.

## 15.3 (Differential): Reserved
## Bytes 204~447 (0x0CC~0x1BF)

Reserved - must be coded as 0x00.

# 16   Annex A.6:  Non-Volatile (NVDIMM-N) Hybrid Memory Parameters (Bytes 192~447, 0x0C0~0x1BF)

## 16.1   Revision 0.1: Byte 192 Coded as 0x01

This clause defines the encoding of SPD bytes 192~447 when Module Type Key Byte 3 contains the following:

- 0x9M, NVDIMM-N
- where M refers to the base memory architecture

NVDIMM types defined at this time are:

| NVDIMM-N | Persistent DRAM using NAND flash |
|---|---|

**See Annex A.0 for common SPD bytes 192~239 (0x0C0~0EF).**

| Module Specific SPD Bytes for NVDIMM-N Module Types | | |
|---|---|---|
| **Byte Number** | | **Function Described** |
| 240 | 0x0F0 | Registering Clock Driver Manufacturer ID Code, First Byte |
| 241 | 0x0F1 | Registering Clock Driver Manufacturer ID Code, Second Byte |
| 242 | 0x0F2 | Register Device Type |
| 243 | 0x0F3 | Register Revision Number |
| 244 | 0x0F4 | Data Buffer Manufacturer ID Code, First Byte |
| 245 | 0x0F5 | Data Buffer Manufacturer ID Code, Second Byte |
| 246 | 0x0F6 | Data Buffer Device Type |
| 247 | 0x0F7 | Data Buffer Revision Number |
| 248~445 | 0x0F8~0x1BD | Reserved |

## 16.2    (NVDIMM-N): Module Specific Device Information
## Bytes 240~247 (0x0F0~0x0F7)

This table lists the SPD byte addresses for the key components on this assembly.

**Table 15 — SPD Byte Addresses for NVDIMM-N**

| Module Specific Device Information | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Device | Manufacturer ID Code, First Byte | | Manufacturer ID Code, Second Byte | | Device Type | | Device Revision | |
| | Dec | Hex | Dec | Hex | Dec | Hex | Dec | Hex |
| Registering Clock Driver (RCD) | 240 | 0x0F0 | 241 | 0x0F1 | 242 | 0x0F2 | 243 | 0x0F3 |
| Data Buffers (DB) | 244 | 0x0F4 | 245 | 0x0F5 | 246 | 0x0F6 | 247 | 0x0F7 |

Manufacturer ID Code: Per JEP106. See SPD bytes 512~513 for an example.

Device types:

| Module Specific Device Types | | | | |
|---|---|---|---|---|
| | Bit 7 | Bit 6 | Bits 5 ~ 4 | Bits 3 ~ 0 |
| Device | Devices Installed | | Reserved | Device Type |
| Registering Clock Driver (RCD) | 0 = Not installed 1 = Installed | Reserved; must be coded as 0 | Reserved. Must be coded as 00 | 0000: DDR5RCD01 (see JESD82-511) All other codes reserved |
| Data Buffers (DB) | 0 = Not installed 1 = Installed | | | 0000: DDR5DB01 (see JESD82-521) All other codes reserved |

Device revision: This byte is coded as a major revision in bits 7~4 and minor revision in bits 3~0, each nibble expressed in BCD. For example, device stepping 3.2 would be coded as 0x32.

## 16.3    (NVDIMM-N): Reserved
## Bytes 248~447 (0x0F8~0x1BF)

Reserved - must be coded as 0x00.

# 17 Annex A.7: Non-Volatile (NVDIMM-P) Hybrid Memory Parameters

## 17.1 (Bytes 192~447, 0x0C0~0x1BF)
## Revision 1.0: Byte 192 Coded as 0x10

This clause defines the encoding of SPD bytes 192~447 when Module Type Key Byte 3 contains the following:

- 0xAM, NVDIMM-P
- where M refers to the base memory architecture

NVDIMM types defined at this time are:

| Type | Definition | Specification |
|---|---|---|
| NVDIMM-P | Transactional credit based memory module | JESD304-5 |

**See Annex A.0 for common SPD bytes 192~239 (0x0C0~0EF).**

| Module Specific SPD Bytes for NVDIMM-P Module Types | | |
|---|---|---|
| **Byte Number** | | **Function Described** |
| 240 | 0x0F0 | Registering Clock Driver Manufacturer ID Code, First Byte |
| 241 | 0x0F1 | Registering Clock Driver Manufacturer ID Code, Second Byte |
| 242 | 0x0F2 | Register Device Type |
| 243 | 0x0F3 | Register Revision Number |
| 244 | 0x0F4 | Data Buffer Manufacturer ID Code, First Byte |
| 245 | 0x0F5 | Data Buffer Manufacturer ID Code, Second Byte |
| 246 | 0x0F6 | Data Buffer Device Type |
| 247 | 0x0F7 | Data Buffer Revision Number |
| 248 | 0x0F8 | Module Storage Capacity, Least Significant Byte |
| 249 | 0x0F9 | Module Storage Capacity, Most Significant Byte |
| 250 | 0x0FA | Protocol Profile |
| 251~445 | 0x0FB~0x1BF | Reserved |

## 17.2 (NVDIMM-P): Module Specific Device Information
## Bytes 240~247 (0x0F0~0x0F7)

This table lists the SPD byte addresses for the key components on this assembly.

### Table 16 — SPD Byte Addresses for NVDIMM-P

| Module Specific Device Information | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Device | Manufacturer ID Code, First Byte | | Manufacturer ID Code, Second Byte | | Device Type | | Device Revision | |
| | Dec | Hex | Dec | Hex | Dec | Hex | Dec | Hex |
| Registering Clock Driver (RCD) | 240 | 0x0F0 | 241 | 0x0F1 | 242 | 0x0F2 | 243 | 0x0F3 |
| Data Buffers (DB) | 244 | 0x0F4 | 245 | 0x0F5 | 246 | 0x0F6 | 247 | 0x0F7 |

Manufacturer ID Code: Per JEP106. See SPD bytes 512~513 for an example.

Device types:

| Module Specific Device Types | | | | |
|---|---|---|---|---|
| | Bit 7 | Bit 6 | Bits 5 ~ 4 | Bits 3 ~ 0 |
| Device | Devices Installed | | Reserved | Device Type |
| Registering Clock Driver (RCD) | 0 = Not installed 1 = Installed | Reserved; must be coded as 0 | Reserved. Must be coded as 00 | 0000: DDR5RCD01 (see JESD82-511) All other codes reserved |
| Data Buffers (DB) | 0 = Not installed 1 = Installed | | | 0000: DDR5DB01 (see JESD82-521) All other codes reserved |

Device revision: This byte is coded as a major revision in bits 7~4 and minor revision in bits 3~0, each nibble expressed in BCD. For example, device stepping 3.2 would be coded as 0x32.

## 17.3 (NVDIMM-P): Module Storage Capacity
## Bytes 248~249 (0x0F8~0x0F9)

These bytes form a 16-bit word which defines the capacity of the non-volatile storage on the NVDIMM-P accessible via the transactional protocol.

| Module Storage Capacity | | |
|---|---|---|
| **Byte 249** | **Byte 248** | |
| **Bits 15~14** | **Bits 13~4** | **Bits 3~0** |
| **Capacity, Base** | **Capacity, Digits** | **Capacity, Tenths** |
| 00 = Megabytes<br>01 = Gigabytes<br>10 = Terabytes<br>11 = Petabytes | Values from 0 to 999<br>All other values reserved | 0000 = 0<br>0001 = 0.1<br>0010 = 0.2<br>0011 = 0.3<br>0100 = 0.4<br>0101 = 0.5<br>0110 = 0.6<br>0111 = 0.7<br>1000 = 0.8<br>1001 = 0.9<br>All other codes reserved |

**Examples:**

| Module Storage Capacity | | |
|---|---|---|
| **Code** | **Hex** | **Meaning** |
| 01 00 1000 0000 0000 | 0x2800 | Module storage capacity = 128 GB |
| 10 00 0000 0001 0000 | 0x8010 | Module storage capacity = 1 TB |
| 10 00 0000 0001 0101 | 0x8015 | Module storage capacity = 1.5 TB |
| 00 11 0010 0000 0000 | 0x3200 | Module storage capacity = 800 MB |

## 17.4 (NVDIMM-P): Protocol Profile
   Byte 250 (0x0FA)

This byte defines the NVDIMM-P profile for the interface protocol. See JESD304-5 for details on profile characteristics.

| Protocol Profile | |
|---|---|
| **Bits 7~4** | **Bits 3~0** |
| **Reserved** | **Protocol Profile** |
| Reserved; must be coded as 0000 | 0000 = NVDIMM-P Profile 0<br>0001 = NVDIMM-P Profile 1<br>All other codes reserved |

## 17.5 (NVDIMM-P): Reserved
   Bytes 251~447 (0x0FB~0x1BF)

Reserved - must be coded as 0x00.

## 17.6    Application Notes for Transactional Credit Based NVDIMMs:

When programming an SPD for use in an NVDIMM supporting this function, the following guidelines must be followed for programming SPD Block 0 to represent the NVDIMM. All other bytes in block 0 are Reserved and must be programmed to 0x00 for use in an NVDIMM.

| Byte | Description | Transactional Credit Based NVDIMM Definition |
|------|-------------|----------------------------------------------|
| 0 | Number of bytes in SPD device | Standard programming |
| 1 | SPD Revision | Standard programming |
| 2 | Key Byte / Host Bus Command Protocol Type | Must be programmed as 20 (0x14), DDR5 NVDIMM-P |
| 3 | Key Byte / Module Type | NVDIMM-P based on RDIMM interface:     Programmed as 0xA1 |
|   |  | NVDIMM-P based on LRDIMM interface:   Programmed as 0xA4 |
| 12 | SDRAM BL32 and Post Package Repair | Must be programmed as 0x00, Burst length of 16, no PPR support |
| 16 | SDRAM Nominal Voltage, VDD | Standard programming |
| 17 | SDRAM Nominal Voltage, VDDQ | Standard programming |
| 20 | SDRAM Minimum Cycle Time ($t_{CKAVG}$min Least Significant Byte | Standard programming |
| 21 | SDRAM Minimum Cycle Time ($t_{CKAVG}$min Most Significant Byte | Standard programming |
| 22 | SDRAM Maximum Cycle Time ($t_{CKAVG}$max Least Significant Byte | Standard programming |
| 23 | SDRAM Maximum Cycle Time ($t_{CKAVG}$max Most Significant Byte | Standard programming |
| 24 | CAS Latencies Supported, First Byte | Standard programming |
| 25 | CAS Latencies Supported, Second Byte | Standard programming |
| 26 | CAS Latencies Supported, Third Byte | Standard programming |
| 27 | CAS Latencies Supported, Fourth Byte | Standard programming |
| 28 | CAS Latencies Supported, Fifth Byte | Standard programming |
| 30 | Minimum CAS Latency Time ($t_{AA}$min) Least Significant Byte | Minimum READ SEND time, $t_{SEND\_RD}$min. |
| 31 | Minimum CAS Latency Time ($t_{AA}$min) Most Significant Byte | Used to calculate $t_{SEND\_RD}$min, $t_{SEND\_WR}$min., $t_{SEND\_SRD}$min. |

Table 17 documents bytes in SPD block 3, Standard Module Parameters – Common SPD Bytes, that must be programmed to specific values to represent the NVDIMM. All other bytes are programmed with standard programming values.

### Table 17 — Bytes in SPD Block 3, Standard Module Parameters

| Byte | Description | Transactional Credit Based NVDIMM Definition |
|------|-------------|----------------------------------------------|
| 235 | Memory Channel Bus Width | Must be 8-bit extension, 32-bits primary data bus |
|   |  | Bits[4~3] = 10 |
|   |  | Bits[2~0] = 010 |

Error logging in the End User blocks of the SPD is not supported in release 1.0 of the NVDIMM-P.

# 18 Blocks 8-9: Manufacturing Information: (Bytes 512~639, 0x200~0x27F)

## 18.1 Byte 512 (0x200): Module Manufacturer ID Code, First Byte
### Byte 513 (0x201): Module Manufacturer ID Code, Second Byte

This two-byte field indicates the manufacturer of the module, and shall be encoded as follows: the first byte is the number of continuation bytes indicated in JEP-106; the second byte is the last non-zero byte of the manufacturer's ID code, again as indicated in JEP-106.

| Byte 513, Bits 7~0 | Byte 512, Bit 7 | Byte 512, Bits 6~0 |
|---|---|---|
| Last non-zero byte, Module Manufacturer | Odd Parity for Byte 512, bits 6~0 | Number of continuation codes, Module Manufacturer |
| See JEP-106 | | See JEP-106 |

**Examples**:

| Company | JEP-106 | | # continuation codes | SPD | |
|---|---|---|---|---|---|
| | Bank | Code | | Byte 512 | Byte 513 |
| Fujitsu | 1 | 04 | 0 | 0x80 | 0x04 |
| US Modular | 5 | A8 | 4 | 0x04 | 0xA8 |

## 18.2 Byte 514 (0x202): Module Manufacturing Location

The module manufacturer shall include an identifier that uniquely defines the manufacturing location of the memory module. While the SPD specification will not attempt to present a decode table for manufacturing sites, the individual manufacturer must keep track of manufacturing location and its appropriate decode represented in this byte.

## 18.3 Bytes 515~516 (0x203~0x204): Module Manufacturing Date

The module manufacturer shall include a date code for the module. The JEDEC definitions for bytes 515 and 516 are year and week respectively. These bytes must be represented in Binary Coded Decimal (BCD). For example, week 47 in year 2014 would be coded as 0x14 (0001 0100) in byte 515 and 0x47 (0100 0111) in byte 516.

## 18.4 Bytes 517~520 (0x205~0x208): Module Serial Number

The supplier shall include a unique serial number for the module. The supplier may use whatever decode method desired to maintain a unique serial number for each module.

One method of achieving this is by assigning a byte in the field from 517~520 as a tester ID byte and using the remaining bytes as a sequential serial number. Bytes 512~520 will then result in a nine-byte unique module identifier. Note that part number is not included in this identifier: the supplier may not give the same value for Bytes 517~520 to more than one DIMM even if the DIMMs have different part numbers.

## 18.5    Bytes 521~550 (0x209~0x226): Module Part Number

The manufacturer's part number shall be included and is written in ASCII format within these bytes. Unused digits are coded as ASCII blanks (0x20).

## 18.6    Byte 551 (0x227): Module Revision Code

This refers to the module revision code. While the SPD specification will not attempt to define the format for this information, the individual manufacturer may keep track of the revision code and its appropriate decode represented in this byte. This revision code refers to the manufacturer's assembly revision level and may be different than the raw card revision in SPD byte 232.

## 18.7    Byte 552 (0x228): DRAM Manufacturer ID Code, First Byte
##        Byte 553 (0x229): DRAM Manufacturer ID Code, Second Byte

This two-byte field indicates the manufacturer of the DRAM on the module, and shall be encoded as follows: the first byte is the number of continuation bytes indicated in JEP-106; the second byte is the last non-zero byte of the manufacturer's ID code, again as indicated in JEP-106.

| Byte 553, Bits 7~0 | Byte 552, Bit 7 | Byte 552, Bits 6~0 |
|---|---|---|
| Last non-zero byte, DRAM Manufacturer | Odd Parity for Byte 552, bits 6~0 | Number of continuation codes, DRAM Manufacturer |
| See JEP-106 | | See JEP-106 |

**Example: See bytes 512~513 for example manufacturer codes.**

## 18.8    Byte 554 (0x22A): DRAM Stepping

This byte defines the vendor die revision level (often called the "stepping") of the DRAMs on the module. This byte is optional. For modules without DRAM stepping information, this byte should be programmed to 0xFF. For DRAM suppliers who use a single letter ("A", "B", etc.) for the stepping indicator, the coding shall be the uppercase ASCII code for that letter.

| Bits 7~0 |
|---|
| **DRAM Stepping** |
| Programmed in straight Hex format - no conversion needed.<br>00 - Valid<br>01 - Valid<br>..<br>FE - Valid<br>FF - Undefined (No Stepping Number Provided) |

**Examples:**

| Code | Meaning |
|---|---|
| 0x00 | Stepping 0 |
| 0x01 | Stepping 1 |
| 0x31 | Stepping 3.1 |
| 0xA3 | Stepping A3 |
| 0xB1 | Stepping B1 |
| 0x41 | Stepping A |
| 0x4A | Stepping J |
| 0xFF | Stepping information not provided |

## 18.9    Bytes 555~639 (0x22B~27F): Manufacturer's Specific Data

The module manufacturer may include any additional information desired into the module within these locations.

# 19 Blocks 10~15: End User Programmable, Bytes 640~1023 (0x280~0x3FF)

These blocks of the SPD are not write protected so that end users may program any values into these bytes during system runtime or for system management.

Optionally, end users may choose to save error logging information in these blocks. If so, the standard error logging format is described below.

## 19.1 Bytes n~n+23: Error Logging Data

Zero or more error logs may appear anywhere in any End User Programmable blocks of the SPD, including over SPD Block boundaries. They may be found by searching for a four byte anchor string. The 20 bytes after the anchor contain information about the error captured. If more than one error log is stored, it is not required to be stored sequentially, nor necessarily at a higher address. The timestamp in the error log may be used to determine the relative order of errors captured.

System BIOSes establish the policy to determine what error types are to be logged, such as only uncorrectable, only correctable, or any error type, and also whether it is the first error found or having exceeding some system specific error count threshold.

Care must be taken not to exceed the write limitations of the SPD device; see JESD300-5 for details.

**Error Logging Data format is as follows:**

| Bytes | Field | Meaning |
|---|---|---|
| n ~ n+3 | Anchor | Anchor String to identify the beginning of an error log |
| n + 4 | Header | Error Type |
| n+5 ~ n+13 | Address | Error Location |
| n+14~n+17 | Location | Timestamp |
| n+18 | Refresh | Highest DRAM Refresh Settings on the Module |
| n+19 ~ n+20 | Temperature | Module Measured Temperature |
| n+21 ~ n+23 | Reserved | Reserved for future use |

Encoding details, by byte:

## 19.2 Bytes n~n+3: Anchor String to Indicate Valid Error Log

Bytes n~n+3 contain the string "_FL_", hex codes 0x5F, 0x46, 0x4C, 0x5F. Parsing the following bytes of the error log and using a "reasonableness" check on the contents may help avoid a false hit on error log data. For example, if the Month field decodes to an invalid Month 15 or the Day field decodes to an invalid Day 0, that indicates a likely false hit on the anchor string.

## 19.3    Byte n+4: Header to Indicate Error Type

Byte n+4 defines the type(s) of errors detected. For DRAM errors, the host system may interrogate the error logging information in each DRAM in the rank where the error occurred to determine the error type(s), and multiple error types may be possible. Other (non-DRAM) errors may be logged in future releases of the SPD document; all other bytes in the error log may be different depending on the coding of byte n+4 bits 7~6.

| Error Type Header | |
|---|---|
| **Bits 7~6** | **Bits 5~0** |
| **Error Log Type** | **Errors Found(s)** |
| 00 = DRAM error log type has been written | Bit code 0 = False, 1 = True<br><br>Bit 0: At least one DRAM uncorrectable error<br>Bit 1: At least one DRAM correctable error<br>Bit 2: At least one DRAM ECS error<br>Bit 3: An error occurred that required an hPPR<br>Bit 4: At least one Post Package Repair (PPR) resource error<br>Bit 5: Reserved; must be coded as 0 |
| Other codes reserved | Coding dependent on Bits 7~6 |
| NOTES:<br>ECS errors may be caused by the row or code word error count (DDR5 SDRAM MR20) or errors per row count (MR19) exceeding an OEM-defined limit.<br>Bits 3 and 4 may be used conjunctively to indicate that an hPPR was necessary and that the required resource was or was not available.<br>PPR Resource error may indicate that the system detected that a row repair was required using sPPR, hPPR, or mPPR, but insufficient PPR resources remained to complete the repair, or that a device dropped below an OEM-defined minimum resource count. This error could also be caused by a guard key error that prevented a requested repair from completing. | |

### 19.3.1  DRAM Error Log Byte Definitions

The following error log bytes are defined when the Error Log Type is 00 (DRAM error log).

It is important to note that conditions described in the DRAM error log capture the state of the memory module at the time that the error log is written. For many reasons, these conditions may or may not apply to the conditions at the time the memory error occurred, such as the difference between when the data was written versus when the data was read. As noted above, users may set error count thresholds before logging, so the log may only represent the last error that occurred. Additional processing is likely required to associate correlation and causation.

## 19.4    Bytes n+5~n+12 (DRAM Error): Error Location

These bytes identify the location(s) where the errors occurred. The error log may only contain information regarding one specific address, however different devices may report different error types regarding that address. The erring devices are identified by a bit map of devices connected on data strobe (DQS) lines.

| Byte | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------|-------|-------|-------|-------|-------|-------|-------|-------|
| | | | | Error Location | | | | |
| n+5 | Reserved | Reserved | CPU2 | CPU1 | CPU0 | Reserved | CPUMC3 | CPUMC2 |
| n+6 | CPUMC1 | CPUMC0 | Reserved | DIMM | CS0_A_n | CS1_A_n | CS0_B_n | CS1_B_n |
| n+7 | Reserved | PAR | CID3/R17 | CID2 | CID1 | CID0 | BG2 | BG1 |
| n+8 | BG0 | BA1 | BA0 | R16 | R15 | R14 | R13 | R12 |
| n+9 | R11 | R10 | R9 | R8 | R7 | R6 | R5 | R4 |
| n+10 | R3 | R2 | R1 | R0 | C10 | C9 | C8 | C7 |
| n+11 | C6 | C5 | C4 | C3 | DQS9A_n | DQS8A_n | DQS7A_n | DQS6A_n |
| n+12 | DQS54_n | DQS4A_n | DQS3A_n | DQS2A_n | DQS1A_n | DQS0A_n | DQS9B_n | DQS8B_n |
| n+13 | DQS7B_n | DQS6B_n | DQS54_n | DQS4B_n | DQS3B_n | DQS2B_n | DQS1B_n | DQS0B_n |

The following data select the location of the SDRAMs and address information for the operation. Note that some signals that are active low (_n) are indicated as active with a 0, inactive with a 1.

Reserved: Must be coded as 0
CPU[2~0]: Identifies the CPU that initiated the operation (0-7)
CPUMC[3~0]: Identifies the memory controller on the CPU that initiated the operation (0-15)
DIMM: Identifies the memory module on the identified memory controller interface (0-1)
     0 = DIMM 0 selected
     1 = DIMM 1 selected
CS[1~0]_A_n: Rank selected on sub-channel A (0-1)
CS[1~0]_B_n: Rank selected on sub-channel B (0-1)
     0 = Rank selected
     1 = Rank not selected
PAR: Parity
CID3/R17: Chip identifier or row address
CID[2~0]: Chip identifier
BG[2~0]: Bank group
BA[1~0]: Bank address
R[16~0]: Row address
C[10:3]: Column address

The following signals are a bit map for every device exhibiting an error for the logged operation, as selected by the data strobes for each device.

DQS[9~0]A_n: Device select (data strobes) on sub-channel A
DQS[9~0]B_n: Device select (data strobes) on sub-channel B
     0 = Device selected
     1 = Device not selected

## 19.5    Bytes n+14~n+17 (DRAM Error): Timestamp

These bytes identify the time at which the error occurred which triggered error logging. The date is per the Gregorian calendar.

| Timestamp | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Byte | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| n+14 | Year | | | | | | Month MSb | |
| n+15 | Month LSb | | Day | | | | | Hour MSb |
| n+16 | Hour LSb | | | Minute MSb | | | | |
| n+17 | Minute LSb | | Second | | | | | |
| **NOTES:** Year is coded as [year - 2020], i.e., the range is from 2020 to 2083 on the Gregorian calendar Month is 1 to 12 Day is 1 to 31 Hour is 0 to 23 Minute is 0 to 59 Second is 0 to 59 | | | | | | | | |

**Example:**

Error logged on March 16, 2025 at 20:15:44

Year = 000101, Month = 0011, Day = 10000, Hour = 10100, Minute = 001111, Second = 101100

Code byte n+14 = 000101 00 (0x14)

Code byte n+15 = 11 10000 1 (0xE1)

Code byte n+16 = 0100 0011 (0x43)

Code byte n+17 = 11 101100 (0xEC)

## 19.6 Byte n+18 (DRAM Error): Highest DRAM Refresh Settings on the Module

This byte saves the state of the highest refresh rate settings of all DRAMs on the module at the time of error log entry, where "highest" is defined by the value of bits OP[2~0] in DRAM mode register MR4. This byte is coded as 0x00 if this information is not being logged. This MR4 information may or may not come from a DRAM reporting an error as indicated in byte n+5 through n+12.

| Highest DRAM Refresh Settings on the Module | | | |
|---|---|---|---|
| Bit 7 | Bit 6 | Bit 5 | Bit 4 |
| Temperature Update Flag (TUF) | Reserved | Wide Range | Refresh tRFC Mode |
| 0 = No change in OP[2~0] since last MR4 read<br>1 = Change in OP[2~0] since last MR4 read | Reserved; must be coded as 0 | 0 = Wide temperature sense range not supported<br>1 = Wide temperature sense range supported | 0 = Normal (tRFC1)<br>1 = Fine granularity (tRFC2) |
| Bit 3 | Bits 2~0 | | |
| Refresh Interval Rate Indicator | Minimum Refresh Rate | | |
| 0 = Not implemented<br>1 = Implemented | If bit 5 = 0<br>  000 = Reserved<br>  001 = tREFI x1 (1x Refresh Rate), t ≤ 80 °C<br>  010 = tREFI x1 (1x Refresh Rate), 80 °C < t ≤ 85 °C<br>  011 = tREFI /2 (2x Refresh Rate), 85 °C < t ≤ 90 °C<br>  100 = tREFI /2 (2x Refresh Rate), 90 °C < t ≤ 95 °C<br>  101 = tREFI /2 (2x Refresh Rate), t > °95 C<br>Other codes reserved | If bit 5 = 1<br>  000 = tREFI x1 (1x Refresh Rate), t ≤ 75 °C<br>  001 = tREFI x1 (1x Refresh Rate), 75 °C < t ≤ 80 °C<br>  010 = tREFI x1 (1x Refresh Rate), 80 °C < t ≤ 85 °C<br>  011 = tREFI /2 (2x Refresh Rate), 85 °C < t ≤ 90 °C<br>  100 = tREFI /2 (2x Refresh Rate), 90 °C < t ≤ 95 °C<br>  101 = tREFI /2 (2x Refresh Rate), 95 °C < t ≤ 100 °C<br>  110 = tREFI /2 (2x Refresh Rate), t > °100 C<br>  111 = Reserved | |
| See the JESD79-5 DDR5 SDRAM definition for mode register MR4.<br>Bit 5, Wide Range, is the same as SPD byte 14 bit 3, included here for simplicity of copying MR4 contents into the error log. | | | |

## 19.7    Byte n+19~20 (DRAM Error): Module Measured Temperature

These bytes store the module temperature measured at the SPD (or ESPD) and external thermal sensors when the error log is created.

| Module Measured Temperature | | |
|---|---|---|
| Byte n+19 | | |
| Bits 7~5 | Bits 4~0 | |
| TS0 Temperature, LSb[1,2,4] | SPD/ESPD Temperature[1,3,4] | |
| Byte n+20 | | |
| Bit 7 | Bits 6~2 | Bits 1~0 |
| Reserved; code as 0 | TS1 Temperature[1,2,4] | TS0 Temperature, MSb[1,2,4] |
| NOTE 1    All thermal sensors are coded as:<br>00000 = Not specified or not installed<br>00001 = 74 °C + 1 = 75 °C or lower<br>00010 = 74 °C + 2 = 76 °C<br>...<br>11110 = 74 °C + 30 = 104 °C<br>11111 = 74 °C + 31 = 105 °C or higher<br>NOTE 2    See the JESD302-1 TS511X Thermal Sensor Specification for mode registers MR49 and MR50:<br>TS temperatures are calculated from TS MR49 and MR50<br>NOTE 3    See the JESD300-5 SPD5118 or JESD316-5 ESPD5216 Device Standards for mode registers MR49 and MR50:<br>SPD temperature is calculated from (E)SPD MR49 and MR50<br>NOTE 4    Devices not installed shall be reported as 00000 | | |

## 19.8    Bytes n+21~n+23 (DRAM Error): Reserved

Must be coded as 0x00 if an error log is stored.

# Standard Improvement Form      JEDEC Standard <u>JESD400-5</u>

The purpose of this form is to provide the Technical Committees of JEDEC with input from the industry regarding usage of the subject standard. Individuals or companies are invited to submit comments to JEDEC. All comments will be collected and dispersed to the appropriate committee(s).

If you can provide input, please complete this form and return to:

JEDEC                             Fax: 703.907.7583
Attn: Publications Department
3103 North 10<sup>th</sup> Street
Suite 240 South
Arlington, VA 22201-2107

1. I recommend changes to the following:

   ☐   Requirement, clause number _____

   ☐   Test method number _____ Clause number _____

   The referenced clause number has proven to be:
   ☐ Unclear   ☐ Too Rigid   ☐ In Error

   ☐ Other _____

2. Recommendations for correction:

   _____
   _____
   _____
   _____

3. Other suggestions for document improvement:

   _____
   _____
   _____
   _____

Submitted by

Name: _____     Phone: _____

Company: _____     E-mail: _____

Address: _____

City/State/Zip: _____     Date: _____