DesignWare® Cores LPDDR5/4/4X Memory Controller

Programming Guide

DWC LPDDR5/4/4X Controller - Product Code: E092-0
DWC LPDDR5/4/4X Controller AFP - Product Code: E093-0
DWC AP LPDDR5/4/4X Controller - Product Code: E094-0

Version 1.10a-lca00
September 2021

Copyright Notice and Proprietary Information

Destination Control Statement

All technical data contained in this publication is subject to the export control laws of the United States of America. Disclosure to nationals of other countries contrary to United States law is prohibited. It is the reader's responsibility to determine the applicable regulations and to comply with them.

Disclaimer

SYNOPSYS, INC., AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Trademarks

Synopsys and certain Synopsys product names are trademarks of Synopsys, as set forth at https://www.synopsys.com/company/legal/trademarks-brands.html

All other product or company names may be trademarks of their respective owners.

Free and Open-Source Software Licensing Notices

If applicable, Free and Open-Source Software (FOSS) licensing notices are available in the product installation.

Third-Party Links

Any links to third-party websites included in this document are for your convenience only. Synopsys does not endorse and is not responsible for such websites and their practices, including privacy practices, availability, and content.

Synopsys, Inc.

www.synopsys.com

Contents

Revision History

The following table provides the history of changes to this Programming Guide:

Version	Date	Description
1.10a-lca00	September 2021	**Added:** ■ "STAR on the Web (SotW)" on page 15 ■ "Synopsys Statement on Inclusivity and Diversity" on page 16 ■ "Special Software Self-Refresh Sequence With Controller Initiated Retraining Followed by Burst PPT2" on page 685 **Updated:** ■ "Register Descriptions" on page 19 ■ "Software Sequences" on page 674
1.01a-lca01	January 2021	**Added:** ■ "Changing Clock Frequencies With Frequency Set Point" ■ "Changing Clock Frequencies With Enabling/Disabling DVFSC" ■ "Changing Clock Frequencies With Enabling/Disabling DVFSQ" ■ "Deep Sleep Mode (DSM) Exit Sequence" **Updated:** ■ "Register Descriptions" ■ "Changing Clock Frequencies Without Frequency Set Point"
1.00a-lca01	June 2020	Initial release

Note In some instances, documentation-only updates occur. The DesignWare IP product https://www.synopsys.com/designware-ip.html has the latest information.

Preface

This Programming Guide describes the software register details, programming sequences, and requirements for the Synopsys DesignWare® Cores DDR Memory Controller (DDRCTL), which is a part of a complete LPDDR5/4/4X interface solution. It also describes the programming flow for various features.

Programming Guide Organization

The chapters of this programming guide are organized as follows:

STAR on the Web (SotW)

You must review all STARs on the Web (SotWs) associated with your product. SotWs are considered a part of the Synopsys documentation suite, and show critical information related to your product. To review product SotWs, refer to the DesignWare IP product information:

https://www.synopsys.com/designware-ip.html

Synopsys Statement on Inclusivity and Diversity

Synopsys is committed to creating an inclusive environment where every employee, customer, and partner feels welcomed. We are reviewing and removing exclusionary language from our products and supporting customer-facing collateral. Our effort also includes internal initiatives to remove biased language from our engineering and working environment, including terms that are embedded in our software and IPs. At the same time, we are working to ensure that our web content and software applications are usable to people of varying abilities. You may still find examples of non-inclusive language in our software or documentation as our IPs implement industry-standard specifications that are currently under review to remove exclusionary language.

Customer Support

To obtain support for your product:

- First, prepare the following debug information, if applicable:
  - For environment setup problems or failures with configuration, simulation, or synthesis that occur within coreConsultant or coreAssembler, use the following menu entry:

File > Build Debug Tar-file

Check all the boxes in the dialog box that apply to your issue. This menu entry gathers all the Synopsys product data needed to begin debugging an issue and writes it to the file <coreTool startup directory>/debug.tar.gz.

- For simulation issues outside of coreConsultant or coreAssembler:
  - Create a waveforms file (such as VPD or VCD).
  - Identify the hierarchy path to the DesignWare instance.
  - Identify the timestamp of any signals or locations in the waveforms that are not understood.
- Then, contact the Support Center, with a description of your question and supplying the previous information, using one of the following methods:
  - For the fastest response, use the SolvNet website. If you fill in your information as explained below, your issue is automatically routed to a support engineer who is experienced with your product. The Sub Product entry is critical for correct routing.

    Go to http://solvnet.synopsys.com/EnterACall and click on the link to open a support case. Provide the requested information, including:

    - Product: DesignWare Cores
    - Sub Product: Memory - Controller
    - Tool Version: 1.10a-lca00
    - Problem Type:
    - Priority:
    - Title: LPDDR5/4/4X: Provide a brief summary of the issue or list the error message you have encountered.
    - Description: For simulation issues, include the timestamp of any signals or locations in waveforms that are not understood.

    After creating the case, attach any debug files you created in the previous step.
  - Or, send an e-mail message to support_center@synopsys.com (your email will be queued and then, on a first-come, first-served basis, manually routed to the correct support engineer):
    - Include the Product name, Sub Product name, process, and Tool Version number in your e-mail (as identified previously) so it can be routed correctly.
    - For simulation issues, include the time stamp of any signals or locations in waveforms that are not understood
    - Attach any debug files you created in the previous step.
  - Or, telephone your local support center:

- North America:
  Call 1-800-245-8005 from 7 AM to 5:30 PM Pacific time, Monday through Friday.
- All other countries:
  https://www.synopsys.com/support/global-support-centers.html

1

Register Descriptions

This chapter details all possible registers in the controller. They are arranged hierarchically into maps and blocks (banks). For configurable IP titles, your actual configuration might not contain all of these registers.

Attention: For configurable IP titles, do not use this document to determine the exact attributes of your register map. It is for reference purposes only.

When you configure the controller in coreConsultant, you must access the register attributes for your actual configuration at `workspace/report/ComponentRegisters.html` or `workspace/report/ComponentRegisters.xml` after you have completed the report creation activity. That report comes from the exact same source as this chapter but removes all the registers that are not in your actual configuration. This does not apply to non-configurable IP titles. In addition, all parameter expressions are evaluated to actual values. Therefore, the Offset and Memory Access values might change depending on your actual configuration.

Some expressions might refer to TCL functions or procedures (sometimes identified as **<functionof>**) that coreConsultant uses to make calculations. The exact formula used by these TCL functions is not provided in this chapter. However, when you configure the controller in coreConsultant, all TCL functions and parameters are evaluated completely; and the resulting values are displayed where appropriate in the coreConsultant GUI reports.

Exists Expressions

These expressions indicate the combination of configuration parameters required for a register, field, or block to exist in the memory map. The expression is only valid in the local context and does not indicate the conditions for existence of the parent. For example, the expression for a bit field in a register assumes that the register exists and does not include the conditions for existence of the register.

Offset

The term *Offset* is synonymous with *Address.*

Memory Access Attributes

The Memory Access attribute is defined as `<ReadBehavior>/<WriteBehavior>` which are defined in the following table.

Table 1-1 Possible Read and Write Behaviors

Read (or Write) Behavior	Description
RC	A read clears this register field.
RS	A read sets this register field.
RM	A read modifies the contents of this register field.
Wo	You can only write once to this register field.
W1C	A write of 1 clears this register field.
W1S	A write of 1 sets this register field.
W1T	A write of 1 toggles this register field.
W0C	A write of 0 clears this register field.
W0S	A write of 0 sets this register field.
W0T	A write of 0 toggles this register field.
WC	Any write clears this register field.
WS	Any write sets this register field.
WM	Any write toggles this register field.
no Read Behavior attribute	You cannot read this register. It is Write-Only.
no Write Behavior attribute	You cannot write to this register. It is Read-Only.

Table 1-2 Memory Access Examples

Memory Access	Description
R	Read-only register field.
W	Write-only register field.
R/W	Read/write register field.
R/W1C	You can read this register field. Writing 1 clears it.
RC/W1C	Reading this register field clears it. Writing 1 clears it.
R/Wo	You can read this register field. You can only write to it once.

Special Optional Attributes

Some register fields might use the following optional attributes.

Table 1-3 Optional Attributes

Attribute	Description
Volatile	As defined by the IP-XACT specification. If true, indicates in the case of a write followed by read, or in the case of two consecutive reads, there is no guarantee as to what is returned by the read on the second transaction or that this return value is consistent with the write or read of the first transaction. The element implies there is some additional mechanism by which this field can acquire new values other than by reads/writes/resets and other access methods known to IP-XACT. For example, when the controller updates the register field contents.
Testable	As defined by the IP-XACT specification. Possible values are unconstrained, untestable, readOnly, writeAsRead, restore. Untestable means that this field is untestable by a simple automated register test. For example, the read-write access of the register is controlled by a pin or another register. readOnly means that you should not write to this register; only read from it. This might apply for a register that modifies the contents of another register.
Reset Mask	As defined by the IP-XACT specification. Indicates that this register field has an unknown reset value. For example, the reset value is set by another register or an input pin; or the register is implemented using RAM.
* Varies	Indicates that the memory access (or reset) attribute (read, write behavior) is not fixed. For example, the read-write access of the register is controlled by a pin or another register. Or when the access depends on some configuration parameter; in this case the post-configuration report in coreConsultant gives the actual access value.

Note: For more information about programming the registers, refer to the "Programming" Chapter.

The Programming Mode Field:

The "Programming Mode" field in the description column of the register tables specifies the type of register. The DDRCTL registers belong to one of the following types:

- Static: Can be written only when the controller is in reset.
- Dynamic: Can be written at any time during operation.
- Quasi Dynamic: Can be written when the controller is in reset and some specific conditions outside reset. There are four groups this type.

Note:

Programming mode describes software responsibility – when software is allowed to update the register field. If software is trying to update the register fields while it is not allowed, the register field may or may not be updated and it can cause unexpected behavior.

For example, Static register may or may not be written outside of reset or SWCTLSTATIC. sw_static_unlock=0, but it is not allowed unless Synopsys explicitly suggests doing that. Similarly, Quasi-dynamic register may or may not be written outside of reset or SWCTL.sw_done=1, but it is not allowed unless Synopsys explicitly suggests doing that.

For more information about the dynamic and quasi dynamic registers, see the **"Programming"** Chapter.

Notes on Timing Registers:

Note 1:

This note refers to register fields whose names include "x32", or "x1024". Computation for these registers are in units of 32, or 1024 clocks.

The DDRCTL contains a timer which issues a pulse once every 32 clock cycles, another which issues a pulse once every 1024 clock cycles. "x32" register fields count pulses of the 32-cycle timer, and "x1024" register fields count pulses of the 1024-cycle timer. These timers are shared by the logic for all of these register fields, and the various periods start without any guaranteed relation to the phase of this timer.

Therefore for "x32" register fields:

- A programmed value of 0 gives 0 clock cycles
- A programmed value of 1 gives a 1 to 32 cycle delay
- A programmed value of 2 gives a 33 to 64 cycle delay and so on

For "x1024" register fields:

- A programmed value of 0 gives no delay
- A programmed value of 1 gives a 1 to 1,024 cycle delay
- A programmed value of 2 gives a 1,025 to 2,048 cycle delay and so on

For minimum delays, the control signal must be set to the smallest number for which the resulting delay is ALWAYS greater than or equal to the required delay; for maximum delays or nominal refresh, this must be set such that the delay is ALWAYS lesser than or equal to the required delay.

This scheme is used to reduce the gate count that would be required to enforce these constraints more precisely.

Note 2

Most of the SDRAM timing registers are in terms of clock cycles. But the value given in an SDRAM datasheet may be in terms of ns/ps. These are primarily minimum timings. Before using them to calculate value to program in the register, it is expected that these are divided by tCK (SDRAM clock period) and rounded up to the next integer value if division does not result in a whole integer value.

Exceptions to this rule are related to maximum timings, where these should be rounded down rather than rounded up, if division does not result in a whole integer value.

These include:

- RFSHSET1TMG0.t_refi_x1_x32
- DRAMSET1TMG0.t_ras_max

Certain SDRAM timing registers clarify this by using the following functions in their descriptions:

- RoundUp (X/Y) means that if X divided by Y does not result in a whole integer value, the result should be rounded up to the next integer value. For example, RoundUp(1000/150) = 7, RoundUp(1050/150) = 7.

- RoundDown (X/Y) means that if X divided by Y does not result in a whole integer value, the result should be rounded down to the next integer value. For example, RoundDown(1000/150) = 6, RoundDown(1050/150) = 7.

The DDRCTL can support up to 4 different sets of frequency sets. Each frequency set is divided in 4 blocks as shown in Figure 1-1.

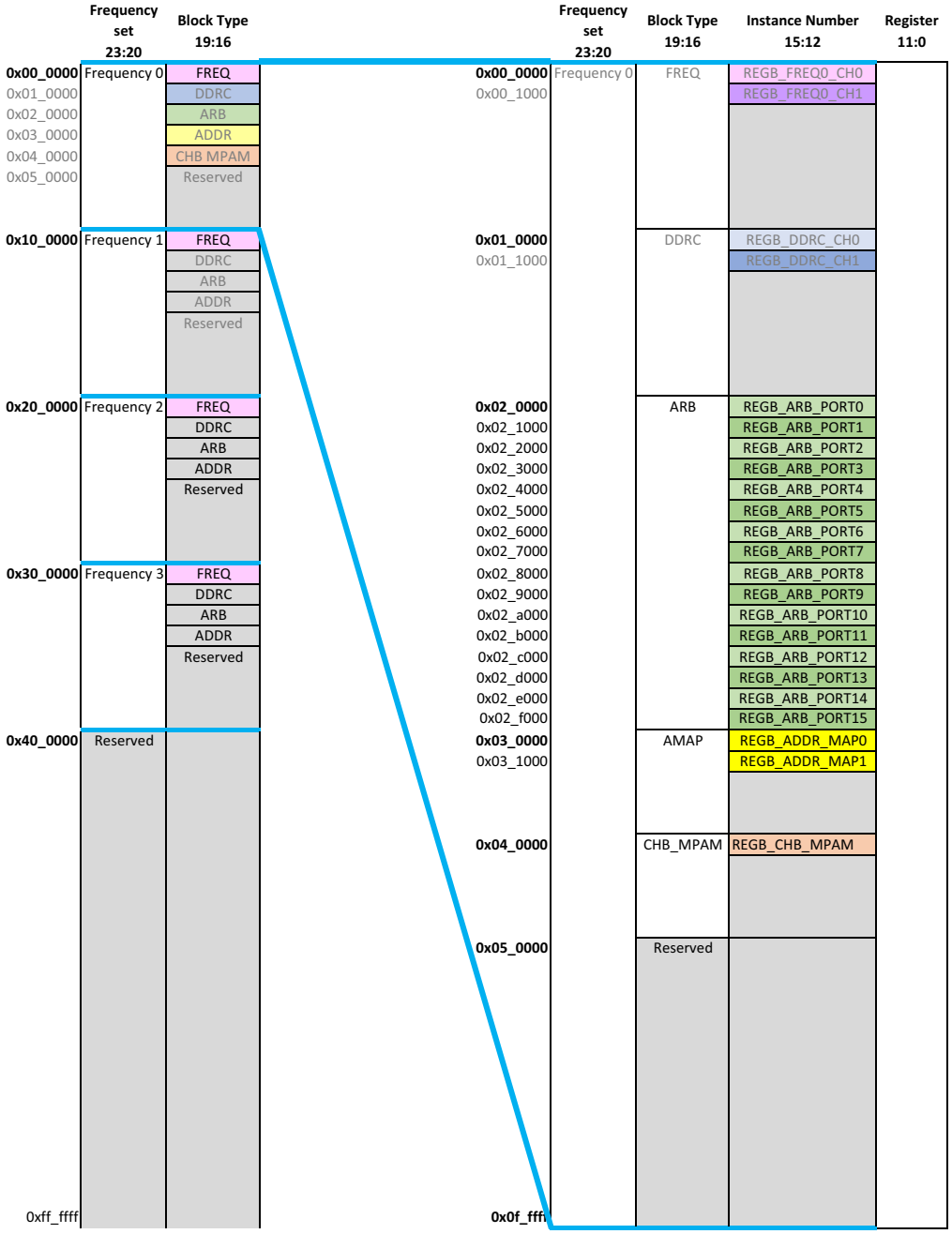
- FREQ block includes registers storing all timings required for correct operations of the controller
- DDRC block has all control registers required by the DDR controller
- ARB block stores XPI port configuration
- ADDR block has registers that define the address map.

The FREQ block is the only block that is currently replicated in every frequency set.

Each region has an equal dedicated address space and internally it can have up to 16 instances of register blocks.

Only instances required for a given configuration are present in the design. For example, a single channel configuration will not have neither REG_FREQ0_CH1 nor REGB_DDRC_CH1. This will ultimately save silicon area when not needed.

Figure 1-1 Memory Map Overview



Register definitions for each component memory map.

Table 1-4 Registers for the DWC_ddrctl_map Memory Map

Register	Offset	Description
Frequency set f Channel c Registers REGB_FREQf_CHc (for f = 0; f <= 3)(for c = 0; c <= 1) Exists: UMCTL2_FREQUENCY_NUM>f && UMCTL2_NUM_DATA_CHANNEL>c		

Register	Offset	Description
“DRAMSET1TMG0” on page 38	0x0+f*0x100000+c*0x1000	SDRAM Timing Register 0 belonging to Timing Set 1
“DRAMSET1TMG1” on page 40	0x4+f*0x100000+c*0x1000	SDRAM Timing Register 1 belonging to Timing Set 1
“DRAMSET1TMG2” on page 42	0x8+f*0x100000+c*0x1000	SDRAM Timing Register 2 belonging to Timing Set 1
“DRAMSET1TMG3” on page 47	0xc+f*0x100000+c*0x1000	SDRAM Timing Register 3 belonging to Timing Set 1
“DRAMSET1TMG4” on page 49	0x10+f*0x100000+c*0x1000	SDRAM Timing Register 4 belonging to Timing Set 1
“DRAMSET1TMG5” on page 51	0x14+f*0x100000+c*0x1000	SDRAM Timing Register 5 belonging to Timing Set 1
“DRAMSET1TMG6” on page 54	0x18+f*0x100000+c*0x1000	SDRAM Timing Register 6 belonging to Timing Set 1
“DRAMSET1TMG7” on page 55	0x1c+f*0x100000+c*0x1000	SDRAM Timing Register 7 belonging to Timing Set 1
“DRAMSET1TMG9” on page 56	0x24+f*0x100000+c*0x1000	SDRAM Timing Register 9 belonging to Timing Set 1
“DRAMSET1TMG12” on page 58	0x30+f*0x100000+c*0x1000	SDRAM Timing Register 12 belonging to Timing Set 1
“DRAMSET1TMG13” on page 59	0x34+f*0x100000+c*0x1000	SDRAM Timing Register 13 belonging to Timing Set 1
“DRAMSET1TMG14” on page 61	0x38+f*0x100000+c*0x1000	SDRAM Timing Register 14 belonging to Timing Set 1
“DRAMSET1TMG17” on page 62	0x44+f*0x100000+c*0x1000	SDRAM Timing Register 17 belonging to Timing Set 1
“DRAMSET1TMG23” on page 63	0x5c+f*0x100000+c*0x1000	SDRAM Timing Register 23 belonging to Timing Set 1
“DRAMSET1TMG24” on page 64	0x60+f*0x100000+c*0x1000	SDRAM Timing Register 24 belonging to Timing Set 1
“DRAMSET1TMG25” on page 66	0x64+f*0x100000+c*0x1000	SDRAM Timing Register 25 belonging to Timing Set 1
“DRAMSET1TMG30” on page 68	0x78+f*0x100000+c*0x1000	SDRAM Timing Register 30 belonging to Timing Set 1
“DRAMSET1TMG32” on page 69	0x80+f*0x100000+c*0x1000	SDRAM Timing Register 32 belonging to Timing Set 1
“INITMR0” on page 71	0x500+f*0x100000+c*0x1000	SDRAM Initialization MR Setting Register 0
“INITMR1” on page 72	0x504+f*0x100000+c*0x1000	SDRAM Initialization MR Setting Register 1

Register	Offset	Description
“INITMR2” on page 73	0x508+f*0x100000+c*0x1000	SDRAM Initialization MR Setting Register 2
“INITMR3” on page 74	0x50c+f*0x100000+c*0x1000	SDRAM Initialization MR Setting Register 3
“DFITMG0” on page 75	0x580+f*0x100000+c*0x1000	DFI Timing Register 0
“DFITMG1” on page 77	0x584+f*0x100000+c*0x1000	DFI Timing Register 1
“DFITMG2” on page 79	0x588+f*0x100000+c*0x1000	DFI Timing Register 2
“DFITMG4” on page 81	0x590+f*0x100000+c*0x1000	DFI Timing Register 4
“DFITMG5” on page 83	0x594+f*0x100000+c*0x1000	DFI Timing Register 5
“DFILPTMG0” on page 85	0x5a0+f*0x100000+c*0x1000	DFI Low Power Timing Register 0
“DFILPTMG1” on page 89	0x5a4+f*0x100000+c*0x1000	DFI Low Power Timing Register 1
“DFIUPDTMG0” on page 91	0x5a8+f*0x100000+c*0x1000	DFI Update Timing Register 0
“DFIUPDTMG1” on page 92	0x5ac+f*0x100000+c*0x1000	DFI Update Timing Register 1
“DFIMSGTMG0” on page 94	0x5b0+f*0x100000+c*0x1000	DFI MC-PHY Message Timing Register 0
“DFIUPDTMG2” on page 95	0x5b4+f*0x100000+c*0x1000	DFI Update Timing Register 2
“RFSHSET1TMG0” on page 97	0x600+f*0x100000+c*0x1000	Refresh Timing Register 0 belonging to Timing Set 1
“RFSHSET1TMG1” on page 101	0x604+f*0x100000+c*0x1000	Refresh Timing Register 1 belonging to Timing Set 1
“RFSHSET1TMG2” on page 103	0x608+f*0x100000+c*0x1000	Refresh Timing Register 2 belonging to Timing Set 1
“RFSHSET1TMG3” on page 104	0x60c+f*0x100000+c*0x1000	Refresh Timing Register 3 belonging to Timing Set 1
“RFSHSET1TMG4” on page 106	0x610+f*0x100000+c*0x1000	Refresh Timing Register 4 belonging to Timing Set 1
“RFSHSET1TMG5” on page 108	0x614+f*0x100000+c*0x1000	Refresh Timing Register 5 belonging to Timing Set 1
“RFMSET1TMG0” on page 110	0x650+f*0x100000+c*0x1000	RFM Timing Register 0 belonging to Timing Set 1
“ZQSET1TMG0” on page 111	0x800+f*0x100000+c*0x1000	ZQ Timing Register 0 belonging to DRAM ZQ timing set 1
“ZQSET1TMG1” on page 112	0x804+f*0x100000+c*0x1000	ZQ Timing Register 1 belonging to DRAM ZQ timing set 1
“DQSOSCCTL0” on page 113	0xa80+f*0x100000+c*0x1000	DQS/WCK Oscillator Control Register 0
“DERATEINT” on page 115	0xb00+f*0x100000+c*0x1000	Temperature Derate Interval Register
“DERATEVAL0” on page 116	0xb04+f*0x100000+c*0x1000	Temperature Derate Timing Register 0

Register	Offset	Description
“DERATEVAL1” on page 118	0xb08+f*0x100000+c*0x1000	Temperature Derate Timing Register 1
“HWLPTMG0” on page 119	0xb80+f*0x100000+c*0x1000	Hardware Low Power Control Register
“SCHEDTMG0” on page 120	0xc00+f*0x100000+c*0x1000	Scheduler Control Register
“PERFHPR1” on page 122	0xc80+f*0x100000+c*0x1000	High Priority Read CAM Register 1
“PERFLPR1” on page 124	0xc84+f*0x100000+c*0x1000	Low Priority Read CAM Register 1
“PERFWR1” on page 125	0xc88+f*0x100000+c*0x1000	Write CAM Register 1
“TMGCFG” on page 126	0xd00+f*0x100000+c*0x1000	Timing Configuration Register
“RANKTMG0” on page 127	0xd04+f*0x100000+c*0x1000	Rank Control Timing 0
“RANKTMG1” on page 130	0xd08+f*0x100000+c*0x1000	Rank Timing Register 1
“PWRTMG” on page 132	0xd0c+f*0x100000+c*0x1000	Low Power Timing Register
DDRC Channel 0 Registers REGB_DDRC_CH0 Exists: Always		
“MSTR0” on page 135	0x10000	Master Register0
“MSTR2” on page 138	0x10008	Master Register2
“MSTR4” on page 139	0x10010	Master Register4
“STAT” on page 141	0x10014	Operating Mode Status Register
“MRCTRL0” on page 144	0x10080	Mode Register Read/Write Control Register 0.
“MRCTRL1” on page 148	0x10084	Mode Register Read/Write Control Register 1
“MRSTAT” on page 149	0x10090	Mode Register Read/Write Status Register
“MRRDATA0” on page 150	0x10094	Mode Register Read Data 0
“MRRDATA1” on page 151	0x10098	Mode Register Read Data 1
“DERATECTL0” on page 152	0x10100	Temperature Derate Control Register 0
“DERATECTL1” on page 154	0x10104	Temperature Derate Control Register 1
“DERATECTL2” on page 156	0x10108	Temperature Derate Control Register 2
“DERATECTL5” on page 158	0x10114	Temperature Derate Control Register 5
“DERATECTL6” on page 160	0x10118	Temperature Derate Control Register 6
“DERATESTAT0” on page 161	0x1011c	Temperature Derate Status Register 0
“DERATEDBGCTL” on page 162	0x10124	Temperature Derate Debug Contrl Register
“DERATEDBGSTAT” on page 163	0x10128	Temperature Derate Debug Status Register
“PWRCTL” on page 165	0x10180	Low Power Control Register

Register	Offset	Description
“HWLPCTL” on page 169	0x10184	Hardware Low Power Control Register
“CLKGATECTL” on page 171	0x1018c	clock gate control
“RFSHMOD0” on page 173	0x10200	Refresh Mode Register 0
“RFSHCTL0” on page 176	0x10208	Refresh Control Register 0
“RFMMOD0” on page 178	0x10220	RFM Mode Register 0
“ZQCTL0” on page 180	0x10280	ZQ Control Register 0
“ZQCTL1” on page 182	0x10284	ZQ Control Register 1
“ZQCTL2” on page 183	0x10288	ZQ Control Register 2
“ZQSTAT” on page 184	0x1028c	ZQ Status Register
“DQSOSCRUNTIME” on page 185	0x10300	DQS/WCK Oscillator Runtime Register
“DQSOSCSTAT0” on page 188	0x10304	DQS/WCK Oscillator Status Register 0
“DQSOSCCFG0” on page 190	0x10308	DQSOSC Config Register 0
“SCHED0” on page 191	0x10380	Scheduler Control Register 0
“SCHED1” on page 197	0x10384	Scheduler Control Register 1
“SCHED3” on page 201	0x1038c	Scheduler Control Register 3
“SCHED4” on page 203	0x10390	Scheduler Control Register 4
“SCHED5” on page 205	0x10394	Scheduler Control Register 5.
“HWFFCCTL” on page 207	0x10400	Hardware Fast Frequency Change (HWFFC) Control Register.
“HWFFCSTAT” on page 210	0x10404	Hardware Fast Frequency Change (HWFFC) Status Register
“DFILPCFG0” on page 212	0x10500	DFI Low Power Configuration Register 0
“DFIUPD0” on page 214	0x10508	DFI Update Register 0
“DFIMISC” on page 216	0x10510	DFI Miscellaneous Control Register
“DFISTAT” on page 219	0x10514	DFI Status Register
“DFIPHYMSTR” on page 220	0x10518	DFI PHY Master
“DFI0MSGCTL0” on page 221	0x10520	DFI0 Message Control Register 0.
“DFI0MSGSTAT0” on page 223	0x10524	DFI0 Message Status Register 0
“POISONCFG” on page 224	0x10580	AXI Poison Configuration Register. Common for all AXI ports
“POISONSTAT” on page 226	0x10584	AXI Poison Status Register

Register	Offset	Description
"ECCCFG0" on page 235	0x10600	ECC Configuration Register 0
"ECCCFG1" on page 239	0x10604	ECC Configuration Register 1
"ECCSTAT" on page 242	0x10608	SECDED ECC Status Register
"ECCCTL" on page 244	0x1060c	ECC Clear Register
"ECCERRCNT" on page 248	0x10610	ECC Error Counter Register
"ECCCADDR0" on page 249	0x10614	ECC Corrected Error Address Register 0
"ECCCADDR1" on page 250	0x10618	ECC Corrected Error Address Register 1
"ECCCSYN0" on page 252	0x1061c	ECC Corrected Syndrome Register 0
"ECCCSYN1" on page 253	0x10620	ECC Corrected Syndrome Register 1
"ECCCSYN2" on page 254	0x10624	ECC Corrected Syndrome Register 2
"ECCBITMASK0" on page 256	0x10628	ECC Corrected Data Bit Mask Register 0
"ECCBITMASK1" on page 257	0x1062c	ECC Corrected Data Bit Mask Register 1
"ECCBITMASK2" on page 258	0x10630	ECC Corrected Data Bit Mask Register 2
"ECCUADDR0" on page 260	0x10634	ECC Uncorrected Error Address Register 0
"ECCUADDR1" on page 261	0x10638	ECC Uncorrected Error Address Register 1
"ECCUSYN0" on page 263	0x1063c	ECC Uncorrected Syndrome Register 0
"ECCUSYN1" on page 264	0x10640	ECC Uncorrected Syndrome Register 1
"ECCUSYN2" on page 265	0x10644	ECC Uncorrected Syndrome Register 2
"ECCPOISONADDR0" on page 267	0x10648	ECC Data Poisoning Address Register 0.
"ECCPOISONADDR1" on page 269	0x1064c	ECC Data Poisoning Address Register 1.
"ECCAPSTAT" on page 270	0x10664	Address protection within ECC Status Register
"OCPARCFG0" on page 271	0x10680	On-Chip Parity Configuration Register 0
"OCPARCFG1" on page 275	0x10684	On-Chip Parity Configuration Register 1
"OCPARSTAT0" on page 277	0x10688	On-Chip Parity Status Register 0
"OCPARSTAT1" on page 283	0x1068c	On-Chip Parity Status Register 1
"OCPARSTAT2" on page 290	0x10690	On-Chip Parity Status Register 2
"OCPARSTAT3" on page 292	0x10694	On-Chip Parity Read Data Log Register 0
"OCPARSTAT4" on page 293	0x10698	On-Chip Parity Write Address Log Register 0
"OCPARSTAT5" on page 294	0x1069c	On-Chip Parity Write Address Log Register 1

Register	Offset	Description
“OCPARSTAT6” on page 295	0x106a0	On-Chip Parity Read Address Log Register 0
“OCPARSTAT7” on page 296	0x106a4	On-Chip Parity Read Address Log Register 1
“OCPARSTAT8” on page 297	0x106a8	On-Chip Parity Read Data Log Register 1
“OCSAPCFG0” on page 298	0x106b0	On-Chip external SRAM Address Protection Configuration Register 0
“OCECCCFG0” on page 300	0x10700	On-Chip ECC Configuration Register 0
“OCECCCFG1” on page 302	0x10704	On-Chip ECC Configuration Register 1
“OCECCSTAT0” on page 305	0x10708	On-Chip ECC Status Register 0
“OCECCSTAT1” on page 307	0x1070c	On-Chip ECC Status Register 1
“OCECCSTAT2” on page 312	0x10710	On-Chip ECC Status Register 2
“OCCAPCFG” on page 313	0x10780	On-Chip command/Address Protection Configuration Register
“OCCAPSTAT” on page 316	0x10784	On-Chip command/Address Protection Status Register
“OCCAPCFG1” on page 318	0x10788	On-Chip command/Address Protection Configuration Register 1
“OCCAPSTAT1” on page 322	0x1078c	On-Chip command/Address Protection Status Register 1
“OCCAPCFG2” on page 325	0x10790	On-Chip command/Address Protection Configuration Register 2
“OCCAPSTAT2” on page 326	0x10794	On-Chip command/Address Protection Status Register 2
“REGPARCFG” on page 327	0x10880	Register Parity Configuration Register
“REGPARSTAT” on page 329	0x10884	Register Parity Status Register
“LNKECCCTL0” on page 330	0x10980	Link-ECC Control Register 0
“LNKECCCTL1” on page 331	0x10984	Link-ECC Control Register 1
“LNKECCPOISONCTL0” on page 333	0x10988	Link-ECC Poison Control Register 0
“LNKECCPOISONSTAT” on page 335	0x1098c	Link-ECC Poison Status Register
“LNKECCINDEX” on page 336	0x10990	Link-ECC Index Register
“LNKECCERRCNT0” on page 337	0x10994	Link-ECC Error Status Register 0
“LNKECCERRSTAT” on page 338	0x10998	Link-ECC Error Status Register 1

Register	Offset	Description
“OPCTRL0” on page 339	0x10b80	Operation Control Register 0
“OPCTRL1” on page 341	0x10b84	Operation Control Register 1
“OPCTRLCAM” on page 342	0x10b88	CAM Operation Control Register
“OPCTRLCMD” on page 345	0x10b8c	Command Operation Control Register
“OPCTRLSTAT” on page 347	0x10b90	Status Operation Control Register
“OPCTRLCAM1” on page 349	0x10b94	CAM Operation Control Register 1
“OPREFCTRL0” on page 350	0x10b98	Refresh Operation Control Register 0
“OPREFCTRL1” on page 361	0x10b9c	Refresh Operation Control Register 1
“OPREFSTAT0” on page 372	0x10ba0	Refresh Operation Status Register 0
“OPREFSTAT1” on page 384	0x10ba4	Refresh Operation Status Register 1
“SWCTL” on page 396	0x10c80	Software Register Programming Control Enable
“SWSTAT” on page 397	0x10c84	Software Register Programming Control Status
“RANKCTL” on page 398	0x10c90	Rank Control Register
“DBICTL” on page 403	0x10c94	DM/DBI Control Register
“ODTMAP” on page 405	0x10c9c	ODT/Rank Map Register
“DATACTL0” on page 408	0x10ca0	Data Control register 0
“SWCTLSTATIC” on page 410	0x10ca4	Static Registers Write Enable
“INITTMG0” on page 411	0x10d00	SDRAM Initialization Timing Register 0
“INITTMG1” on page 413	0x10d04	SDRAM Initialization Timing Register 1
“PPT2CTRL0” on page 414	0x10f00	PPT2 Control Register
“PPT2STAT0” on page 416	0x10f10	Status PPT2 Control Register
“DDRCTL_VER_NUMBER” on page 418	0x10ff8	DDRCTL Version Number Register
“DDRCTL_VER_TYPE” on page 419	0x10ffc	DDRCTL Version Type Register
DDRC Channel 1 Registers REGB_DDRC_CH1 Exists: UMCTL2_DUAL_DATA_CHANNEL==1		
“MSTR4” on page 421	0x11010	Master Register4
“STAT” on page 423	0x11014	Operating Mode Status Register
“MRCTRL0” on page 426	0x11080	Mode Register Read/Write Control Register 0.

Register	Offset	Description
“MRCTRL1” on page 430	0x11084	Mode Register Read/Write Control Register 1
“MRSTAT” on page 431	0x11090	Mode Register Read/Write Status Register
“MRRDATA0” on page 432	0x11094	Mode Register Read Data 0
“MRRDATA1” on page 433	0x11098	Mode Register Read Data 1
“DERATECTL5” on page 434	0x11114	Temperature Derate Control Register 5
“DERATESTAT0” on page 436	0x1111c	Temperature Derate Status Register 0
“DERATEDBGCTL” on page 437	0x11124	Temperature Derate Debug Contrl Register
“DERATEDBGSTAT” on page 438	0x11128	Temperature Derate Debug Status Register
“PWRCTL” on page 440	0x11180	Low Power Control Register
“HWLPCTL” on page 444	0x11184	Hardware Low Power Control Register
“ZQCTL1” on page 445	0x11284	ZQ Control Register 1
“ZQSTAT” on page 446	0x1128c	ZQ Status Register
“DQSOSCSTAT0” on page 447	0x11304	DQS/WCK Oscillator Status Register 0
“HWFFCSTAT” on page 449	0x11404	Hardware Fast Frequency Change (HWFFC) Status Register
“DFISTAT” on page 451	0x11514	DFI Status Register
“DFI0MSGCTL0” on page 452	0x11520	DFI0 Message Control Register 0.
“DFI0MSGSTAT0” on page 454	0x11524	DFI0 Message Status Register 0
“ECCSTAT” on page 455	0x11608	SECDED ECC Status Register
“ECCCTL” on page 457	0x1160c	ECC Clear Register
“ECCERRCNT” on page 461	0x11610	ECC Error Counter Register
“ECCCADDR0” on page 462	0x11614	ECC Corrected Error Address Register 0
“ECCCADDR1” on page 463	0x11618	ECC Corrected Error Address Register 1
“ECCCSYN0” on page 465	0x1161c	ECC Corrected Syndrome Register 0
“ECCCSYN1” on page 466	0x11620	ECC Corrected Syndrome Register 1
“ECCCSYN2” on page 467	0x11624	ECC Corrected Syndrome Register 2
“ECCBITMASK0” on page 469	0x11628	ECC Corrected Data Bit Mask Register 0
“ECCBITMASK1” on page 470	0x1162c	ECC Corrected Data Bit Mask Register 1
“ECCBITMASK2” on page 471	0x11630	ECC Corrected Data Bit Mask Register 2
“ECCUADDR0” on page 473	0x11634	ECC Uncorrected Error Address Register 0
“ECCUADDR1” on page 474	0x11638	ECC Uncorrected Error Address Register 1

Register	Offset	Description
"ECCUSYN0" on page 476	0x1163c	ECC Uncorrected Syndrome Register 0
"ECCUSYN1" on page 477	0x11640	ECC Uncorrected Syndrome Register 1
"ECCUSYN2" on page 478	0x11644	ECC Uncorrected Syndrome Register 2
"ECCAPSTAT" on page 480	0x11664	Address protection within ECC Status Register
"OCPARCFG0" on page 481	0x11680	On-Chip Parity Configuration Register 0
"OCPARSTAT2" on page 483	0x11690	On-Chip Parity Status Register 2
"OCCAPCFG1" on page 485	0x11788	On-Chip command/Address Protection Configuration Register 1
"OCCAPSTAT1" on page 489	0x1178c	On-Chip command/Address Protection Status Register 1
"OCCAPCFG2" on page 492	0x11790	On-Chip command/Address Protection Configuration Register 2
"OCCAPSTAT2" on page 493	0x11794	On-Chip command/Address Protection Status Register 2
"LNKECCCTL0" on page 494	0x11980	Link-ECC Control Register 0
"LNKECCCTL1" on page 495	0x11984	Link-ECC Control Register 1
"LNKECCPOISONCTL0" on page 497	0x11988	Link-ECC Poison Control Register 0
"LNKECCPOISONSTAT" on page 499	0x1198c	Link-ECC Poison Status Register
"LNKECCINDEX" on page 500	0x11990	Link-ECC Index Register
"LNKECCERRCNT0" on page 501	0x11994	Link-ECC Error Status Register 0
"LNKECCERRSTAT" on page 502	0x11998	Link-ECC Error Status Register 1
"OPCTRL1" on page 503	0x11b84	Operation Control Register 1
"OPCTRLCAM" on page 504	0x11b88	CAM Operation Control Register
"OPCTRLCMD" on page 507	0x11b8c	Command Operation Control Register
"OPCTRLSTAT" on page 509	0x11b90	Status Operation Control Register
"OPCTRLCAM1" on page 511	0x11b94	CAM Operation Control Register 1
"OPREFCTRL0" on page 512	0x11b98	Refresh Operation Control Register 0
"OPREFCTRL1" on page 523	0x11b9c	Refresh Operation Control Register 1
"OPREFSTAT0" on page 534	0x11ba0	Refresh Operation Status Register 0
"OPREFSTAT1" on page 546	0x11ba4	Refresh Operation Status Register 1

Register	Offset	Description
"DBICTL" on page 558	0x11c94	DM/DBI Control Register
"ODTMAP" on page 560	0x11c9c	ODT/Rank Map Register
"INITTMG0" on page 563	0x11d00	SDRAM Initialization Timing Register 0
"INITTMG1" on page 565	0x11d04	SDRAM Initialization Timing Register 1
Arbiter Port p Registers REGB_ARB_PORTp (for p = 0; p <= 15) Exists: UMCTL2_A_NPORTS>p && UMCTL2_INCL_ARB_OR_CHB==1		
"PCCFG" on page 567	0x20000+p*0x1000	Port Common Configuration Register.
"PCFGR" on page 568	0x20004+p*0x1000	Configuration Read Register
"PCFGW" on page 571	0x20008+p*0x1000	Configuration Write Register
"PCFGIDMASKCHc (for c = 0; c <= 15)" on page 574	0x20010+p*0x1000+c*0x8	Channel c Configuration ID mask register
"PCFGIDVALUECHc (for c = 0; c <= 15)" on page 575	0x20014+p*0x1000+c*0x8	Channel c Configuration ID value register
"PCTRL" on page 576	0x20090+p*0x1000	Port Control Register
"PCFGQOS0" on page 577	0x20094+p*0x1000	Port n Read QoS Configuration Register 0
"PCFGQOS1" on page 580	0x20098+p*0x1000	Port n Read QoS Configuration Register 1
"PCFGWQOS0" on page 581	0x2009c+p*0x1000	Port n Write QoS Configuration Register 0
"PCFGWQOS1" on page 583	0x200a0+p*0x1000	Port n Write QoS Configuration Register 1
"SARBASEs (for s = 0; s <= 3)" on page 584	0x200c0+p*0x1000+s*0x8	SAR Base Address Register s.
"SARSIZEs (for s = 0; s <= 3)" on page 585	0x200c4+p*0x1000+s*0x8	SAR Size Register s.
"SBRCTL" on page 586	0x200e0+p*0x1000	Scrubber Control Register
"SBRSTAT" on page 591	0x200e4+p*0x1000	Scrubber Status Register
"SBRWDATA0" on page 593	0x200e8+p*0x1000	Scrubber Write Data Pattern0
"SBRWDATA1" on page 594	0x200ec+p*0x1000	Scrubber Write Data Pattern1
"SBRSTART0" on page 595	0x200f0+p*0x1000	Scrubber Start Address Mask Register 0
"SBRSTART1" on page 596	0x200f4+p*0x1000	Scrubber Start Address Mask Register 1
"SBRRANGE0" on page 597	0x200f8+p*0x1000	Scrubber Address Range Mask Register 0
"SBRRANGE1" on page 598	0x200fc+p*0x1000	Scrubber Address Range Mask Register 1
"SBRSTART0DCH1" on page 599	0x20100+p*0x1000	Scrubber Start Address Mask Register 0 for Data Channel 1

Register	Offset	Description
“SBRSTART1DCH1” on page 600	0x20104+p*0x1000	Scrubber Start Address Mask Register 1 for Data Channel 1
“SBRRANGE0DCH1” on page 601	0x20108+p*0x1000	Scrubber Address Range Mask Register 0 for Data Channel 1
“SBRRANGE1DCH1” on page 602	0x2010c+p*0x1000	Scrubber Address Range Mask Register 1 for Data Channel 1
“PDCH” on page 603	0x20110+p*0x1000	Port Data Channel
“PSTAT” on page 607	0x20114+p*0x1000	Port Status Register
Address Map 0 Registers REGB_ADDR_MAP0 Exists: Always		
“ADDRMAP0” on page 612	0x30000	Address Map Register 0
“ADDRMAP1” on page 613	0x30004	Address Map Register 1
“ADDRMAP3” on page 615	0x3000c	Address Map Register 3
“ADDRMAP4” on page 617	0x30010	Address Map Register 4
“ADDRMAP5” on page 619	0x30014	Address Map Register 5
“ADDRMAP6” on page 623	0x30018	Address Map Register 6
“ADDRMAP7” on page 625	0x3001c	Address Map Register 7
“ADDRMAP8” on page 627	0x30020	Address Map Register 8
“ADDRMAP9” on page 629	0x30024	Address Map Register 9
“ADDRMAP10” on page 631	0x30028	Address Map Register 10
“ADDRMAP11” on page 633	0x3002c	Address Map Register 11
“ADDRMAP12” on page 634	0x30030	Address Map Register 12
“ADDRMAPLUTCFG” on page 635	0x30080	Addr CS map LUT config register
“ADDRMAPLUTCTRL” on page 638	0x30084	Addr CS map LUT control register
“ADDRMAPLUTRDATA” on page 640	0x30088	Addr CS map LUT read data register
Address Map 1 Registers REGB_ADDR_MAP1 Exists: UMCTL2_HET_RANK==1		
“ADDRMAP1” on page 642	0x31004	Address Map Register 1
“ADDRMAP3” on page 644	0x3100c	Address Map Register 3

Register	Offset	Description
“ADDRMAP4” on page 646	0x31010	Address Map Register 4
“ADDRMAP5” on page 648	0x31014	Address Map Register 5
“ADDRMAP6” on page 652	0x31018	Address Map Register 6
“ADDRMAP7” on page 654	0x3101c	Address Map Register 7
“ADDRMAP8” on page 656	0x31020	Address Map Register 8
“ADDRMAP9” on page 658	0x31024	Address Map Register 9
“ADDRMAP10” on page 660	0x31028	Address Map Register 10
“ADDRMAP11” on page 662	0x3102c	Address Map Register 11

1.1 REGB_FREQf_CHc Registers

This register block contains registers related to timings in the DDRC controller. Registers shared by all blocks are only duplicated for every channel, hence they only reside in REGB_FREQ0_CHc.

1.1.1 DRAMSET1TMG0

- **Description:** SDRAM Timing Register 0 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x0+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

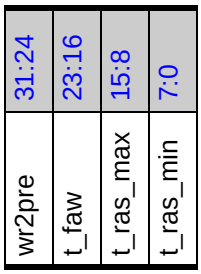


Table 1-5 Fields for Register: DRAMSET1TMG0

Bits	Name	Memory Access	Description
31:24	wr2pre	R/W	Minimum time between write and precharge to same bank. Specifications: WL + BL/2 + tWR where: ■ WL = write latency ■ BL = burst length. This must match the value programmed in the BL bit of the mode register to the SDRAM. BST (burst terminate) is not supported at present. ■ tWR = Write recovery time. This comes directly from the SDRAM specification. Add one extra cycle for LPDDR4/5 for this parameter. For DDR5, add one extra cycle when CRCPARCTL1.wr_crc_enable = 1. Note that, depending on the PHY, if using LRDIMM, it may be necessary to adjust the value of this parameter to compensate for the extra cycle of latency through the LRDIMM. Unit: DRAM clock cycles. **Value After Reset:** 0xf **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 2, Group 4

Bits	Name	Memory Access	Description
23:16	t_faw	R/W	tFAW: At most 4 banks must be activated in a rolling window of tFAW cycles. Unit: DRAM clock cycles. **Value After Reset:** 0x10 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
15:8	t_ras_max	R/W	tRAS(max): Maximum time between activate and precharge to same bank. This is the maximum time that a page can be kept open t_ras_max must be set to RoundDown(tRAS(max)/tCK/1024). Unit: 1024 DRAM clock cycles. **Value After Reset:** 0x1b **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
7:0	t_ras_min	R/W	tRAS(min): Minimum time between activate and precharge to the same bank. Unit: DRAM clock cycles. **Value After Reset:** 0xf **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

1.1.2 DRAMSET1TMG1

- **Description:** SDRAM Timing Register 1 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x4+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

Table 1-6 Fields for Register: DRAMSET1TMG1

Bits	Name	Memory Access	Description
31:22			**Reserved Field:** Yes
21:16	t_xp	R/W	tXP: Minimum time after power-down exit to any operation. ■ DDR4 (C/A parity not enabled): tXP ■ DDR4 (C/A parity enabled): (tXP+PL) ■ DDR5: tXP ■ DDR5 (L)RDIMM: max (tXP, tRPDX) ■ LPDDR4 (tCKELPD is defined in spec): larger of tXP and tCKELPD instead. ■ LPDDR4 (tCKELPD is not defined in spec): tXP. ■ LPDDR5 (MR17.OP[4] ODTD-CS is enabled): max (tXP, tPDXCSODTON) ■ LPDDR5 (MR17.OP[4] ODTD-CS is disabled): tXP Unit: DRAM clock cycles. **Value After Reset:** 0x8 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

Bits	Name	Memory Access	Description
15:8	rd2pre	R/W	tRTP: Minimum time from read to precharge of same bank. ■ DDR4: Max of following two equations: tAL + max (RoundUp(tRTP/tCK), 4) or, RL + BL/2 - tRP (*). ■ DDR5: tRTP ■ LPDDR4 - BL/2 + max(RoundUp(tRTP/tCK),8) - 8 ■ LPDDR5(BG mode): BL/n_min + RU(tRBTP/tCK) ■ LPDDR5(16B mode): BL/n + RU(tRBTP/tCK) (*) When both DDR4 SDRAM and ST-MRAM are used simultaneously, use SDRAM's tRP value for calculation. Unit: DRAM clock cycles. **Value After Reset:** 0x4 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 2, Group 4
7:0	t_rc	R/W	tRC: Minimum time between activates to same bank. Unit: DRAM clock cycles. **Value After Reset:** 0x14 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

1.1.3 DRAMSET1TMG2

- **Description:** SDRAM Timing Register 2 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x8+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

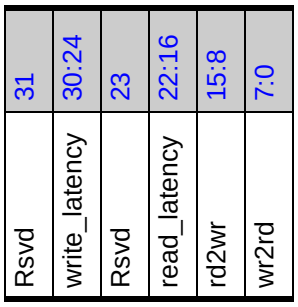


Table 1-7 Fields for Register: DRAMSET1TMG2

Bits	Name	Memory Access	Description
31			**Reserved Field:** Yes
30:24	write_latency	R/W	Set to WL Time from write command to write data on SDRAM interface. This must be set to WL. Note that, depending on the PHY, if using RDIMM/LRDIMM, it may be necessary to adjust the value of WL to compensate for the extra cycle of latency through the RDIMM/LRDIMM. For all protocols, in addition to programming this register field, it is necessary to program DFITMG0 and DFITMG1 to control the read and write latencies Unit: DRAM clock cycles. **Value After Reset:** 0x3 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 2, Group 4
23			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
22:16	read_latency	R/W	Set to RL Time from read command to read data on SDRAM interface. This must be set to RL. Note that, depending on the PHY, if using RDIMM/LRDIMM, it may be necessary to adjust the value of RL to compensate for the extra cycle of latency through the RDIMM/LRDIMM. In addition to programming this register field, it is necessary to program DFITMG0 and DFITMG1 to control the read and write latencies Unit: DRAM clock cycles. **Value After Reset:** 0x5 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 2, Group 4

Bits	Name	Memory Access	Description
15:8	rd2wr	R/W	■ DDR4: RL + BL/2 + 1 + WR_PREAMBLE - WL ■ DDR5: CL - CWL + BL/2 + 2 - (Read DQS offset) + (RD_POSTAMBLE-0.5) + WR_PREAMBLE ■ LPDDR4(DQ ODT is Disabled): RL + BL/2 + RU(tDQSCKmax/tCK) + WR_PREAMBLE + RD_POSTAMBLE - WL ■ LPDDR4(DQ ODT is Enabled) : RL + BL/2 + RU(tDQSCKmax/tCK) + RD_POSTAMBLE - ODTLon - RD(tODTon(min)/tCK) + 1 ■ LPDDR5 (BG mode && DQ ODT is Disabled): RL + BL/n_max + RU(tWCKDQO(max)/tCK) - WL ■ LPDDR5 (BG mode && DQ ODT is Enabled) : RL + BL/n_max + RU(tWCKDQO(max)/tCK) + RD(tRPST/tCK) - ODTLon - RD(tODTon(min)/tCK) + 1 ■ LPDDR5 (16B mode && DQ ODT is Disabled): RL + BL/n + RU(tWCKDQO(max)/tCK) - WL ■ LPDDR5 (16B mode && DQ ODT is Enabled) : RL + BL/n + RU(tWCKDQO(max)/tCK) + RD(tRPST/tCK) - ODTLon - RD(tODTon(min)/tCK) + 1 Minimum time from read command to write command. Include time for bus turnaround and all per-bank, per-rank, and global constraints. Please see the relevant PHY databook for details of what must be included here. Where: ■ WL = write latency ■ BL = burst length. This must match the value programmed in the BL bit of the mode register to the SDRAM ■ RL = read latency = CAS latency ■ WR_PREAMBLE = 1 (1tCK write preamble), 2 (2tCK write preamble). ■ RD_POSTAMBLE = 0.5 (0.5tCK read postamble), 1.5 (1.5tCK read postamble). After PHY has completed training the value programmed may need to be increased. Refer to relevant PHY documentation.

Bits	Name	Memory Access	Description
15:8...(cont.)	rd2wr.	R/W	For LPDDR4, if derating is enabled (DERATECTL0.derate_enable=1), derated tDQSCKmax must be used. Note that, depending on the PHY, if using LRDIMM, it may be necessary to adjust the value of this parameter to compensate for the extra cycle of latency through the LRDIMM. Unit: DRAM clock cycles. **Value After Reset:** 0x6 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 2, Group 4

Bits	Name	Memory Access	Description
7:0	wr2rd	R/W	■ DDR4: CWL + PL + BL/2 + tWTR_L ■ DDR5: CWL + BL/2 + tWTR_L ■ LPDDR4: WL + BL/2 + tWTR + 1 ■ LPDDR5(BG mode): WL + BL/n_max + RU(tWTR_L/tCK) ■ LPDDR5(16B mode): WL + BL/n + RU(tWTR/tCK) In DDR4, minimum time from write command to read command for same bank group. In others, minimum time from write command to read command. Includes time for bus turnaround, recovery times, and all per-bank, per-rank, and global constraints. Please see the relevant PHY databook for details of what must be included here. Where: ■ CWL = CAS write latency ■ WL = Write latency ■ PL = Parity latency ■ BL = burst length. This must match the value programmed in the BL bit of the mode register to the SDRAM ■ tWTR_L = internal write to read command delay for same bank group. This comes directly from the SDRAM specification. ■ tWTR = internal write to read command delay. This comes directly from the SDRAM specification. After PHY has completed training the value programmed may need to be increased. Refer to relevant PHY documentation. Add one extra cycle for LPDDR4 operation. WTR_L must be increased by one if DDR4 2tCK write preamble is used. Unit: DRAM clock cycles. **Value After Reset:** 0xd **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 2, Group 4

1.1.4 DRAMSET1TMG3

- **Description:** SDRAM Timing Register 3 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0xc+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

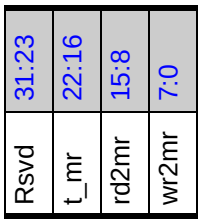


Table 1-8 Fields for Register: DRAMSET1TMG3

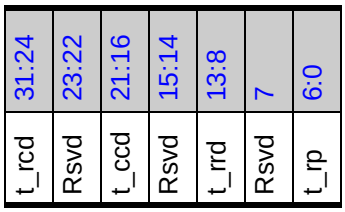
Bits	Name	Memory Access	Description
31:23			**Reserved Field:** Yes
22:16	t_mr	R/W	Time from MRW/MRS to valid command ■ DDR4: Set this to the larger of tMOD + AL and tMRD. If C/A parity is enabled, tMOD_PAR(tMOD+PL) + AL and tMRD_PAR(tMOD+PL) and used instead. If CAL mode is enabled, tCAL must be added to the above. Note that if using RDIMM/LRDIMM, depending on the PHY, it may be necessary to adjust the value of this parameter to compensate for the extra cycle of latency applied to mode register writes by the RDIMM/LRDIMM chip. Also note that if using LRDIMM, the minimum value of this register is tMRD_L2. ■ DDR5: Set this to the larger of tMRR, tMRW, tMRWPD, tMRD and tMPC_DELAY. ■ LPDDR4:Set this to the larger of tMRR, tMRW, tMRWCKEL and tMRD. ■ LPDDR5:Set this to the larger of (tMRR + tMRW) and tMRD. Unit: DRAM clock cycles. **Value After Reset:** 0x4 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

Bits	Name	Memory Access	Description
15:8	rd2mr	R/W	Time from Read to MRW/MRR command. ■ LPDDR4: RL + BL/2 + RU(tDQSCKmax/tCK) + RD(tRPST) + max(RU(7.5ns/tCK),8nCK) + nRTP - 8 ■ LPDDR5: RL + RU(tWCKDQO(max)/tCK)) + BL/n_max + MAX[RU(7.5ns/tCK),4nCK] + nRBTP Unit: DRAM clock cycles. **Value After Reset:** 0x4 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
7:0	wr2mr	R/W	Time from Write to MRW/MRR command. ■ LPDDR4: WL + 1 + BL/2 + max(RU(7.5ns/tCK),8nCK) + nWR ■ LPDDR5: WL + BL/n_max + MAX[RU(7.5ns/tCK),4nCK] + nWR Unit: DRAM clock cycles. **Value After Reset:** 0x4 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

1.1.5 DRAMSET1TMG4

- **Description:** SDRAM Timing Register 4 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x10+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.



31:24	23:22	21:16	15:14	13:8	7	6:0
t_rcd	Rsvd	t_ccd	Rsvd	t_rrd	Rsvd	t_rp

Table 1-9 Fields for Register: DRAMSET1TMG4

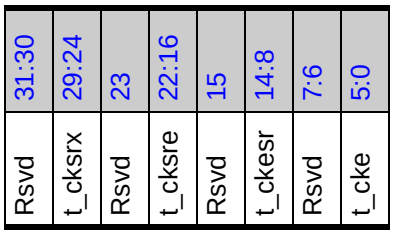
Bits	Name	Memory Access	Description
31:24	t_rcd	R/W	tRCD - tAL: Minimum time from activate to read or write command to same bank. Note: For DDR5 2N mode, it is recommended to set this value as multiple of MEMC_FREQ_RATIO to improve the performance. Unit: DRAM clock cycles. **Value After Reset:** 0x5 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 2, Group 4
23:22			**Reserved Field:** Yes
21:16	t_ccd	R/W	This is the minimum time between two reads or two writes. ■ DDR4: tCCD_L ■ LPDDR4: tCCD ■ LPDDR5: BL/n Don't Care for DDR5 (see DRAMSET1TMG26.t_ccd_r/t_ccd_w in DDR5). Unit: DRAM clock cycles. **Value After Reset:** 0x4 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
15:14			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
13:8	t_rrd	R/W	For DDR4/DDR5/LPDDR5(BG mode): Minimum time between activates from bank "a" to bank "b" for same bank group. For LPDDR4/LPDDR5(16B mode): Minimum time between activates from bank "a" to bank "b". ■ DDR4/5: tRRD_L ■ LPDDR4: RU(tRRD/tCK) ■ LPDDR5(BG mode): RU(tRRD_L/tCK) ■ LPDDR5(16B mode): RU(tRRD/tCK) Unit: DRAM clock cycles. **Value After Reset:** 0x4 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
7			**Reserved Field:** Yes
6:0	t_rp	R/W	tRP: Minimum time from single-bank precharge to activate of same bank. t_rp must be set to RoundUp(tRP/tCK). Unit: DRAM clock cycles. **Value After Reset:** 0x5 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

1.1.6 DRAMSET1TMG5

- **Description:** SDRAM Timing Register 5 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x14+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.



31:30	29:24	23	22:16	15	14:8	7:6	5:0
Rsvd	t_cksrx	Rsvd	t_cksre	Rsvd	t_ckesr	Rsvd	t_cke

Table 1-10 Fields for Register: DRAMSET1TMG5

Bits	Name	Memory Access	Description
31:30			**Reserved Field:** Yes
29:24	t_cksrx	R/W	This is the time before Self Refresh Exit that CK is maintained as a valid clock before issuing SRX. Specifies the clock stable time before SRX. Recommended settings: ■ LPDDR4: tCKCKEH ■ LPDDR5: tCKCSH ■ DDR4: tCKSRX ■ DDR5: tCKSRX Unit: DRAM clock cycles. **Value After Reset:** 0x5 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
23			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
22:16	t_cksre	R/W	This is the time after Self Refresh Down Entry/Power Down Entry that CK is maintained as a valid clock. Specifies the clock disable delay after SRE/PDE. Recommended settings: ■ LPDDR4: tCKELCK ■ LPDDR5: tCSLCK ■ DDR4: tCKSRE (+ PL(parity latency)(*)) ■ DDR5: tCKLCS (*)Only if CRCPARCTL1.caparity_disable_before_sr=0, this register must be increased by PL. Unit: DRAM clock cycles. **Value After Reset:** 0x5 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
15			**Reserved Field:** Yes
14:8	t_ckesr	R/W	Minimum CKE low width for Self refresh or Self refresh power down entry to exit timing in memory clock cycles. Recommended settings: ■ LPDDR4: max(tCKE, tSR) ■ LPDDR5: tSR ■ DDR4: tCKESR (+ PL(parity latency)(*)) ■ DDR5: Don't care (*)Only if CRCPARCTL1.caparity_disable_before_sr=0, this register must be increased by PL. Unit: DRAM clock cycles. **Value After Reset:** 0x4 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
7:6			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
5:0	t_cke	R/W	Delay time between PDE and PDX. ■ LPDDR4: tCKE ■ LPDDR5: tCSPD ■ DDR4: tPD (+ PL(parity latency)(*)) ■ DDR5: Don't care (*)Only if CRCPARCTL1.caparity_disable_before_sr=0, this register must be increased by PL. Unit: DRAM clock cycles. **Value After Reset:** 0x3 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

1.1.7 DRAMSET1TMG6

- **Description:** SDRAM Timing Register 6 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x18+f*0x100000+c*0x1000
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_FREQf_CHc.
Note: This register is only applicable for designs supporting LPDDR4/LPDDR5 SDRAM memories

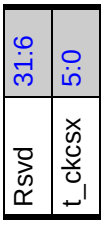


Table 1-11 Fields for Register: DRAMSET1TMG6

Bits	Name	Memory Access	Description
31:6			**Reserved Field:** Yes
5:0	t_ckcsx	R/W	This is the time before Clock Stop Exit that CK is maintained as a valid clock before issuing Clock Stop Exit. Specifies the clock stable time before next command after Clock Stop Exit. Recommended settings: ■ LPDDR4/5: tXP + 2 This is only present for designs supporting LPDDR devices. Unit: DRAM clock cycles. **Value After Reset:** 0x5 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

1.1.8 DRAMSET1TMG7

- **Description:** SDRAM Timing Register 7 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x1c+f*0x100000+c*0x1000
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_FREQf_CHc.
Note: This register is only applicable for designs supporting LPDDR4/LPDDR5 SDRAM memories

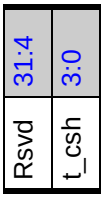


Table 1-12 Fields for Register: DRAMSET1TMG7

Bits	Name	Memory Access	Description
31:4			**Reserved Field:** Yes
3:0	t_csh	R/W	CS High Pulse width at PDX ■ LPDDR5: tCSH Unit: DRAM clock cycles. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

1.1.9 DRAMSET1TMG9

- **Description:** SDRAM Timing Register 9 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x24+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

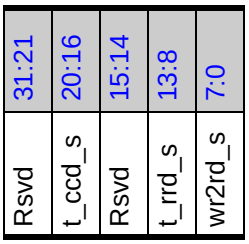


Table 1-13 Fields for Register: DRAMSET1TMG9

Bits	Name	Memory Access	Description
31:21			**Reserved Field:** Yes
20:16	t_ccd_s	R/W	tCCD_S: This is the minimum time between two reads or two writes for different bank group. For bank switching (from bank "a" to bank "b"), the minimum time is this value + 1. Note: This register field is only applicable for designs supporting DDR4/LPDDR5 SDRAM memories. Unit: DRAM clock cycles. **Value After Reset:** 0x4 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
15:14			**Reserved Field:** Yes
13:8	t_rrd_s	R/W	tRRD_S: Minimum time between activates from bank "a" to bank "b" for different bank group. Note: This register field is only applicable for designs supporting DDR4/DDR5/LPDDR5 SDRAM memories. Unit: DRAM clock cycles. **Value After Reset:** 0x4 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

Bits	Name	Memory Access	Description
7:0	wr2rd_s	R/W	Minimum time from write command to read command for different bank group. Includes time for bus turnaround, recovery times, and all per-bank, per-rank, and global constraints. Applicable only in designs configured to support DDR SDRAM memories or LPDDR5 SDRAM memories. **DDR4/DDR5 designs:**CWL + PL + BL/2 + tWTR_S Minimum time from write command to read command for different bank group. Includes time for bus turnaround, recovery times, and all per-bank, per-rank, and global constraints. Where: ■ CWL = CAS write latency ■ PL = Parity latency ■ BL = burst length. This must match the value programmed in the BL bit of the mode register to the SDRAM ■ tWTR_S = internal write to read command delay for different bank group. This comes directly from the SDRAM specification. WTR_S must be increased by one if DDR4 2tCK write preamble is used. **LPDDR5 designs:**WL + BL/n_min +RU(tWTR_S/tCK) Where: ■ WL = Write Latency ■ BL/n_min = Effective Burst Length ■ tWTR_S = internal write to read command delay for different bank group. Unit: DRAM clock cycles. **Value After Reset:** 0xd **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 2, Group 4

1.1.10 DRAMSET1TMG12

- **Description:** SDRAM Timing Register 12 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x30+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

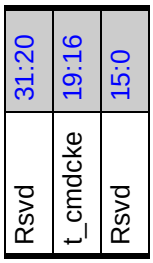


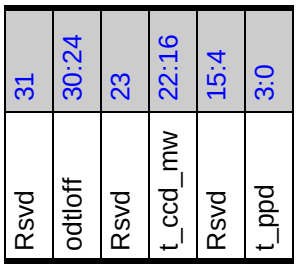
Table 1-14 Fields for Register: DRAMSET1TMG12

Bits	Name	Memory Access	Description
31:20			**Reserved Field:** Yes
19:16	t_cmdcke	R/W	tCMDCKE: Delay from valid command to PDE ■ LPDDR4: max(tESCKE, tCMDCKE) ■ LPDDR5: max(tESPD, tCMDPD) Unit: DRAM clock cycles. **Value After Reset:** 0x2 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
15:0			**Reserved Field:** Yes

1.1.11 DRAMSET1TMG13

- **Description:** SDRAM Timing Register 13 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x34+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.



31	30:24	23	22:16	15:4	3:0
Rsvd	odtloff	Rsvd	t_ccd_mw	Rsvd	t_ppd

Table 1-15 Fields for Register: DRAMSET1TMG13

Bits	Name	Memory Access	Description
31			**Reserved Field:** Yes
30:24	odtloff	R/W	LPDDR4: ODTLoff: This is the latency from CAS-2 command to tODToff reference. LPDDR5: ODTLoff: This is the latency from the Write or Mask Write command (the rising edge of the clock) to tODToff reference. Unit: DRAM clock cycles. **Value After Reset:** 0x1c **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
23			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
22:16	t_ccd_mw	R/W	This is the minimum time from write or masked write to masked write command for same bank. ■ LPDDR4: tCCDMW ■ LPDDR5(BG mode): 4*BL/n_max ■ LPDDR5(16B mode): 4*BL/n Note: This register field is only applicable for designs supporting LPDDR4/LPDDR5 SDRAM memories. Unit: DRAM clock cycles. **Value After Reset:** 0x20 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
15:4			**Reserved Field:** Yes
3:0	t_ppd	R/W	LPDDR4/5 and DDR5: tPPD: This is the minimum time from precharge to precharge command. Note: This register is not applicable for DDR4 SDRAM memories. Unit: DRAM clock cycles. **Value After Reset:** 0x4 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

1.1.12 DRAMSET1TMG14

- **Description:** SDRAM Timing Register 14 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x38+f*0x100000+c*0x1000
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_FREQf_CHc.

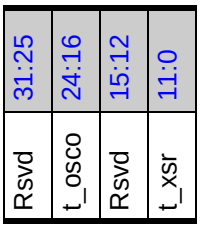


Table 1-16 Fields for Register: DRAMSET1TMG14

Bits	Name	Memory Access	Description
31:25			**Reserved Field:** Yes
24:16	t_osco	R/W	Minimum time from DQS Oscillator (LPDDR4) or WCK2DQI/WCK2DQO Oscillator(LPDDR5) stop to Mode register readout. LPDDR4 : tOSCO = max(40ns,8nCK) LPDDR5 : tOSCODQI = tOSCODQO = max(40ns,8nCK) Unit: DRAM clock cycles. **Value After Reset:** 0x8 **Exists:** LPDDR45_DQSOSC_EN==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
15:12			**Reserved Field:** Yes
11:0	t_xsr	R/W	tXSR: Exit Self Refresh to any command. The value 0xfff is illegal for this register field. Unit: DRAM clock cycles. **Value After Reset:** 0xa0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

1.1.13 DRAMSET1TMG17

- **Description:** SDRAM Timing Register 17 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x44+f*0x100000+c*0x1000
- **Exists:** UMCTL2_HWFFC_EN==1

This register is in block REGB_FREQf_CHc.

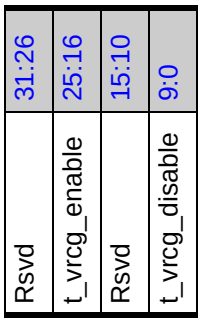


Table 1-17 Fields for Register: DRAMSET1TMG17

Bits	Name	Memory Access	Description
31:26			**Reserved Field:** Yes
25:16	t_vrcg_enable	R/W	LPDDR4: tVRCG_ENABLE: VREF high current mode enable time. Unit: DRAM clock cycles. **Value After Reset:** 0x0 **Exists:** UMCTL2_HWFFC_EN==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 4
15:10			**Reserved Field:** Yes
9:0	t_vrcg_disable	R/W	LPDDR4: tVRCG_ENABLE: VREF high current mode disable time. Unit: DRAM clock cycles. **Value After Reset:** 0x0 **Exists:** UMCTL2_HWFFC_EN==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 4

1.1.14 DRAMSET1TMG23

- **Description:** SDRAM Timing Register 23 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x5c+f*0x100000+c*0x1000
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_FREQf_CHc.
Note: This register is only applicable for designs supporting LPDDR5 SDRAM memories

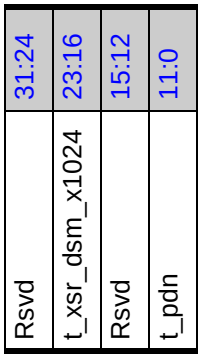


Table 1-18 Fields for Register: DRAMSET1TMG23

Bits	Name	Memory Access	Description
31:24			**Reserved Field:** Yes
23:16	t_xsr_dsm_x1024	R/W	Delay from Deep Sleep Mode Exit to SRX. Unit: Multiples of 1024 DRAM clock cycles. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic - Refresh Related
15:12			**Reserved Field:** Yes
11:0	t_pdn	R/W	Minimum interval between Deep Sleep Mode Entry and Exit. Unit: DRAM clock cycles. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic - Refresh Related

1.1.15 DRAMSET1TMG24

- **Description:** SDRAM Timing Register 24 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x60+f*0x100000+c*0x1000
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_FREQf_CHc.
Note: This register is only applicable for designs supporting LPDDR5 SDRAM memories

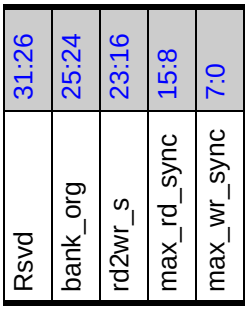


Table 1-19 Fields for Register: DRAMSET1TMG24

Bits	Name	Memory Access	Description
31:26			**Reserved Field:** Yes
25:24	bank_org	R/W	Select Bank/ Bank group organization: ■ 00: 4 Banks/ 4 Bank groups ■ 01: 8 Banks (Reserved) ■ 10: 16 Banks ■ 11: Reserved **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 2, Group 4

Bits	Name	Memory Access	Description
23:16	rd2wr_s	R/W	Minimum time from read command to write command for different bank group. Includes time for bus turnaround, recovery times and all per-bank, per-rank and global constraints. ■ LPDDR5(DQ ODT is disabled): RL + BL/n_min + RU(tWCKDQO(max)/tCK) - WL ■ LPDDR5(DQ ODT is enabled): RL + BL/n_min + RU(tWCKDQO(max)/tCK) + RD(tRPST/tCK) - ODTLon - RD(tODTon(min)/tCK) Unit: DRAM clock cycles. **Value After Reset:** 0xf **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 2, Group 4
15:8	max_rd_sync	R/W	Minimum time from read command to WCK2CK sync OFF. RL + BL/n_max + RD(tWCKPST/tCK) Unit: DRAM clock cycles. **Value After Reset:** 0xf **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 2, Group 4
7:0	max_wr_sync	R/W	Minimum time from write command to WCK2CK sync OFF. WL + BL/n_max + RD(tWCKPST/tCK) Unit: DRAM clock cycles. **Value After Reset:** 0xf **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 2, Group 4

1.1.16 DRAMSET1TMG25

- **Description:** SDRAM Timing Register 25 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x64+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

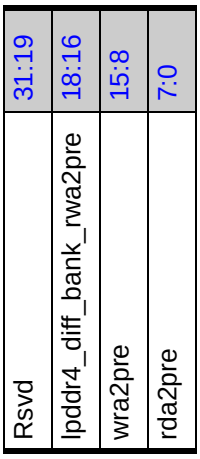


Table 1-20 Fields for Register: DRAMSET1TMG25

Bits	Name	Memory Access	Description
31:19			**Reserved Field:** Yes
18:16	lpddr4_diff_bank_rwa2pre	R/W	Set the timing constraint between different bank RD/WR/MWR/ACT and PRE in LPDDR4. ■ LPDDR4 JESD209-4A requires 4 cycles ■ LPDDR4 JESD209-4B requires 2 cycles Value of 1, 3, 5, 6, and 7 are illegal. Don't care for LPDDR5. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 2, Group 4

Bits	Name	Memory Access	Description
15:8	wra2pre	R/W	Time between write with AP and precharge to same bank. ■ LPDDR4: WL + BL/2 + nWR + 1 ■ LPDDR5: WL + BL/n_min + nWR + 1 ■ DDR4: WL + BL/2 + WR Don't care for DDR5. Unit: DRAM clock cycles. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 2, Group 4
7:0	rda2pre	R/W	Time between read with AP and precharge to same bank. ■ LPDDR4: nRTP ■ LPDDR5: BL/n_min + nRBTP ■ DDR4: RTP Don't care for DDR5. Unit: DRAM clock cycles. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 2, Group 4

1.1.17 DRAMSET1TMG30

- **Description:** SDRAM Timing Register 30 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x78+f*0x100000+c*0x1000
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_FREQf_CHc.
Note: This register is only applicable for designs supporting LPDDR5 SDRAM memories

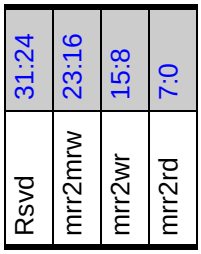


Table 1-21 Fields for Register: DRAMSET1TMG30

Bits	Name	Memory Access	Description
31:24			**Reserved Field:** Yes
23:16	mrr2mrw	R/W	MRR to MRW delay Unit: DRAM clock cycles. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
15:8	mrr2wr	R/W	MRR to WR delay Unit: DRAM clock cycles. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
7:0	mrr2rd	R/W	MRR to RD delay Unit: DRAM clock cycles. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

1.1.18 DRAMSET1TMG32

- **Description:** SDRAM Timing Register 32 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x80+f*0x100000+c*0x1000
- **Exists:** DDRCTL_LPDDR==1 && MEMC_NUM_RANKS_GT_1==1

This register is in block REGB_FREQf_CHc.
Note: This register is only applicable for designs supporting LPDDR5 SDRAM memories

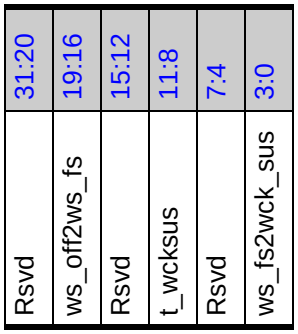


Table 1-22 Fields for Register: DRAMSET1TMG32

Bits	Name	Memory Access	Description
31:20			**Reserved Field:** Yes
19:16	ws_off2ws_fs	R/W	Delay from CAS-WS_OFF to next CAS_WS-FS command. Unit: DRAM clock cycles. **Value After Reset:** 0x3 **Exists:** DDRCTL_LPDDR==1 && MEMC_NUM_RANKS_GT_1==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
15:12			**Reserved Field:** Yes
11:8	t_wcksus	R/W	tWCKSUS: Delay from CAS-WCK_SUSPEND command to next READ/WRITE/MASK WRITE command. Unit: DRAM clock cycles. **Value After Reset:** 0x4 **Exists:** DDRCTL_LPDDR==1 && MEMC_NUM_RANKS_GT_1==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
7:4			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
3:0	ws_fs2wck_sus	R/W	Delay from CAS-WS_FS command to next CAS-WCK_SUSPEND command. Unit: DRAM clock cycles. **Value After Reset:** 0x8 **Exists:** DDRCTL_LPDDR==1 && MEMC_NUM_RANKS_GT_1==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

1.1.19 INITMR0

- **Description:** SDRAM Initialization MR Setting Register 0
- **Size:** 32 bits
- **Offset:** 0x500+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

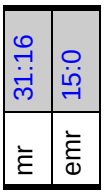


Table 1-23 Fields for Register: INITMR0

Bits	Name	Memory Access	Description
31:16	mr	R/W	■ DDR4: Value loaded into MR0 register. ■ DDR5: Don't care ■ LPDDR4: Value to write to MR1 register ■ LPDDR5: Don't care (For LPDDR5, DDRCTL doesn't send MRW commands to initialize MR in the SDRAM even if INITTMG0.skip_dram_init=0) **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 4
15:0	emr	R/W	■ DDR4: Value to write to MR1 register Set bit 7 to 0. ■ DDR5: Don't care ■ LPDDR4: Value to write to MR2 register ■ LPDDR5: Don't care (For LPDDR5, DDRCTL doesn't send MRW commands to initialize MR in the SDRAM even if INITTMG0.skip_dram_init=0) **Value After Reset:** 0x510 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 4

1.1.20 INITMR1

- **Description:** SDRAM Initialization MR Setting Register 1
- **Size:** 32 bits
- **Offset:** 0x504+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

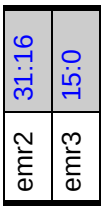


Table 1-24 Fields for Register: INITMR1

Bits	Name	Memory Access	Description
31:16	emr2	R/W	■ DDR4: Value to write to MR2 register ■ DDR5: Don't care ■ LPDDR4: Value to write to MR3 register ■ LPDDR5: Don't care (For LPDDR5, DDRCTL doesn't send MRW commands to initialize MR in the SDRAM even if INITTMG0.skip_dram_init=0) **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 4
15:0	emr3	R/W	■ DDR4: Value to write to MR3 register ■ DDR5: Don't care ■ LPDDR4: Value to write to MR13 register ■ LPDDR5: Don't care (For LPDDR5, DDRCTL doesn't send MRW commands to initialize MR in the SDRAM even if INITTMG0.skip_dram_init=0) **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

1.1.21 INITMR2

- **Description:** SDRAM Initialization MR Setting Register 2
- **Size:** 32 bits
- **Offset:** 0x508+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

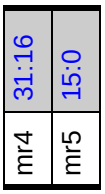


Table 1-25 Fields for Register: INITMR2

Bits	Name	Memory Access	Description
31:16	mr4	R/W	■ DDR4: Value to be loaded into SDRAM MR4 registers. ■ DDR5: Don't care ■ LPDDR4: Value to be loaded into SDRAM MR11 registers (not applicable for initialization, but this is used when HWFFC is performed) ■ LPDDR5: Don't care (For LPDDR5, DDRCTL doesn't send MRW commands to initialize MR in the SDRAM even if INITTMG0.skip_dram_init=0) **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
15:0	mr5	R/W	■ DDR4: Value to be loaded into SDRAM MR5 registers. ■ DDR5: Don't care ■ LPDDR4: Value to be loaded into SDRAM MR12 registers (not applicable for initialization, but this is used when HWFFC is performed) ■ LPDDR5: Don't care (For LPDDR5, DDRCTL doesn't send MRW commands to initialize MR in the SDRAM even if INITTMG0.skip_dram_init=0) **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 4

1.1.22 INITMR3

- **Description:** SDRAM Initialization MR Setting Register 3
- **Size:** 32 bits
- **Offset:** 0x50c+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

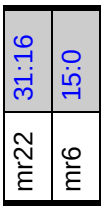


Table 1-26 Fields for Register: INITMR3

Bits	Name	Memory Access	Description
31:16	mr22	R/W	■ LPDDR4 Value to be loaded into SDRAM MR22 registers (not applicable for initialization, but this is used when HWFFC is performed) ■ LPDDR5: Don't care (For LPDDR5, DDRCTL doesn't send MRW commands to initialize MR in the SDRAM even if INITTMG0.skip_dram_init=0) **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 4
15:0	mr6	R/W	■ DDR4 Value to be loaded into SDRAM MR6 registers. ■ DDR5: Don't care ■ LPDDR4 Value to be loaded into SDRAM MR14 registers (not applicable for initialization, but this is used when HWFFC is performed) ■ LPDDR5: Don't care (For LPDDR5, DDRCTL doesn't send MRW commands to initialize MR in the SDRAM even if INITTMG0.skip_dram_init=0) **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 4

1.1.23 DFITMG0

- **Description:** DFI Timing Register 0
- **Size:** 32 bits
- **Offset:** 0x580+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

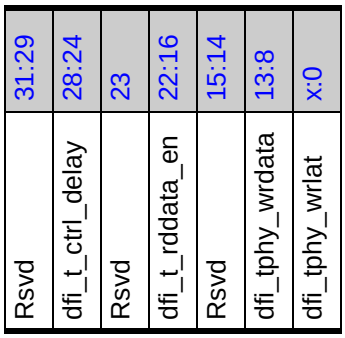


Table 1-27 Fields for Register: DFITMG0

Bits	Name	Memory Access	Description
31:29			**Reserved Field:** Yes
28:24	dfi_t_ctrl_delay	R/W	Specifies the number of DFI clock cycles after an assertion or de-assertion of the DFI control signals that the control signals at the PHY-DRAM interface reflect the assertion or de-assertion. If the DRAM clock and the memory clock are not phase-aligned, this timing parameter must be rounded up to the next integer value. Note that if using RDIMM/LRDIMM, it is necessary to increment this parameter by RDIMM's/LRDIMM's extra cycle of latency in terms of DFI clock. Unit: DFI clock cycles. **Value After Reset:** 0x7 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 4
23			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
22:16	dfi_t_rddata_en	R/W	Time from the assertion of a read command on the DFI interface to the assertion of the dfi_rddata_en signal. Refer to PHY specification for correct value. This corresponds to the DFI parameter trddata_en. Note that, depending on the PHY, if using RDIMM/LRDIMM, it may be necessary to use the adjusted value of CL in the calculation of trddata_en. This is to compensate for the extra cycle(s) of latency through the RDIMM/LRDIMM. Unit: DRAM data clock cycles. **Value After Reset:** 0x2 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 4
15:14			**Reserved Field:** Yes
13:8	dfi_tphy_wrdata	R/W	Specifies the number of clock cycles between when dfi_wrdata_en is asserted to when the associated write data is driven on the dfi_wrdata signal. This corresponds to the DFI timing parameter tphy_wrdata. Refer to PHY specification for correct value. Note, max supported value is 8. Unit: DRAM data clock cycles. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 4
x:0	dfi_tphy_wrlat	R/W	Write latency Number of clocks from the write command to write data enable (dfi_wrdata_en). This corresponds to the DFI timing parameter tphy_wrlat. Refer to PHY specification for correct value. For DDR5/4, note that, depending on the PHY, if using RDIMM/LRDIMM, it may be necessary to use the adjusted value of CL in the calculation of tphy_wrlat. This is to compensate for the extra cycle(s) of latency through the RDIMM/LRDIMM. For LPDDR4, dfi_tphy_wrlat>60 is not supported. Unit: DRAM data clock cycles. **Value After Reset:** 0x2 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4 **Range Variable[x]:** "(DDRCTL_DDR_DUAL_CHANNEL_EN==1) ? 7 : 6" - 1

1.1.24 DFITMG1

- **Description:** DFI Timing Register 1
- **Size:** 32 bits
- **Offset:** 0x584+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

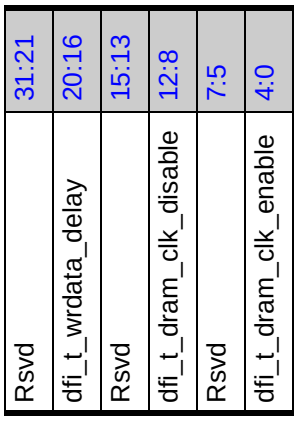


Table 1-28 Fields for Register: DFITMG1

Bits	Name	Memory Access	Description
31:21			**Reserved Field:** Yes
20:16	dfi_t_wrdata_delay	R/W	Specifies the number of DFI clock cycles between when the dfi_wrdata_en signal is asserted and when the corresponding write data transfer is completed on the DRAM bus. This corresponds to the DFI timing parameter twrdata_delay. For LPDDR5, this should be set to "twck_delay + BL/n_max - BL/n_min" instead of twrdata_delay. twck_delay specifies the time from dfi_wck_en deassertion to when WCK transfer completes on the DRAM bus and is defined by the PHY Refer to PHY specification for correct value. When TMGCFG.frequency_ratio is set to 0(1:2 Mode), divided the value by 2 and round it up to the next integer value. When TMGCFG.frequency_ratio is set to 1(1:4 Mode), divided the value by 4 and round it up to the next integer value. Unit: DFI clock cycles. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 4
15:13			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
12:8	dfi_t_dram_clk_disable	R/W	Specifies the number of DFI clock cycles from the assertion of the dfi_dram_clk_disable signal on the DFI until the clock to the DRAM memory devices, at the PHY-DRAM boundary, maintains a low value. If the DRAM clock and the memory clock are not phase aligned, this timing parameter must be rounded up to the next integer value. Unit: DFI clock cycles. Note: For SNPS DDR54 PHY, the dfi_t_dram_clk_disable should always be equal to dfi_t_ctrl_delay. Please see the PHY databook. **Value After Reset:** 0x4 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 4
7:5			**Reserved Field:** Yes
4:0	dfi_t_dram_clk_enable	R/W	Specifies the number of DFI clock cycles from the de-assertion of the dfi_dram_clk_disable signal on the DFI until the first valid rising edge of the clock to the DRAM memory devices, at the PHY-DRAM boundary. If the DRAM clock and the memory clock are not phase aligned, this timing parameter must be rounded up to the next integer value. Unit: DFI clock cycles. Note: For SNPS DDR54 PHY, the dfi_t_dram_clk_enable should always be equal to dfi_t_ctrl_delay. Please see the PHY databook. **Value After Reset:** 0x4 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 4

1.1.25 DFITMG2

- **Description:** DFI Timing Register 2
- **Size:** 32 bits
- **Offset:** 0x588+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

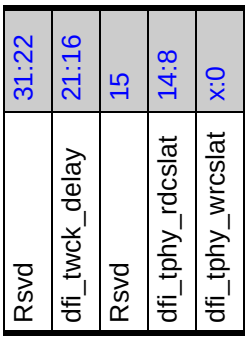


Table 1-29 Fields for Register: DFITMG2

Bits	Name	Memory Access	Description
31:22			**Reserved Field:** Yes
21:16	dfi_twck_delay	R/W	Number of DFI PHY clock cycles from dfi_wck_en is de-asserted to when the WCK transfer completes on th DRAM bus. Refer to PHY specification for correct value. Unit: DFI PHY clock cycles. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
15			**Reserved Field:** Yes
14:8	dfi_tphy_rdcslat	R/W	Number of DFI PHY clock cycles between when a read command is sent on the DFI control interface and when the associated dfi_rddata_cs signal is asserted. This corresponds to the DFI timing parameter tphy_rdcslat. Refer to PHY specification for correct value. Unit: DRAM data clock cycles. **Value After Reset:** 0x2 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

Bits	Name	Memory Access	Description
x:0	dfi_tphy_wrcslat	R/W	Number of DFI PHY clock cycles between when a write command is sent on the DFI control interface and when the associated dfi_wrdata_cs signal is asserted. This corresponds to the DFI timing parameter tphy_wrcslat. Refer to PHY specification for correct value. Unit: DRAM data clock cycles. **Value After Reset:** 0x2 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4 **Range Variable[x]:** "(DDRCTL_DDR_DUAL_CHANNEL_EN==1) ? 7 : 6" - 1

1.1.26 DFITMG4

- **Description:** DFI Timing Register 4
- **Size:** 32 bits
- **Offset:** 0x590+f*0x100000+c*0x1000
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_FREQf_CHc.
Note: This register is only applicable for designs supporting LPDDR5 SDRAM memories

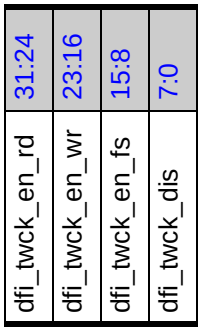


Table 1-30 Fields for Register: DFITMG4

Bits	Name	Memory Access	Description
31:24	dfi_twck_en_rd	R/W	Defines the number of clocks between the CAS_WS_RD command to when the dfi_wck_en signal is driven. Unit: WCK cycles **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 4
23:16	dfi_twck_en_wr	R/W	Defines the number of clocks between the CAS_WS_WR command to when the dfi_wck_en signal is driven. Unit: WCK cycles **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 4
15:8	dfi_twck_en_fs	R/W	Defines the number of clocks between the CAS_WS_FS command to when the dfi_wck_en signal is driven. Unit: WCK cycles **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 && MEMC_NUM_RANKS>1 **Programming Mode:** Quasi-dynamic Group 1, Group 4

Bits	Name	Memory Access	Description
7:0	dfi_twck_dis	R/W	Defines the number of clock cycles between the last command (LAST CMD) without a WCK synchronization required (assuming no command issued) or any command that disables the WCK to when the dfi_wck_en signal is disabled. Unit: WCK cycles **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 4

1.1.27 DFITMG5

- **Description:** DFI Timing Register 5
- **Size:** 32 bits
- **Offset:** 0x594+f*0x100000+c*0x1000
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_FREQf_CHc.
Note: This register is only applicable for designs supporting LPDDR5 SDRAM memories

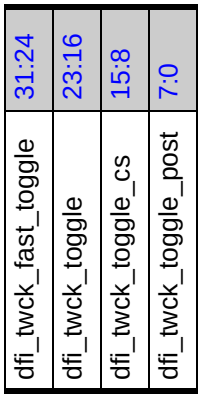


Table 1-31 Fields for Register: DFITMG5

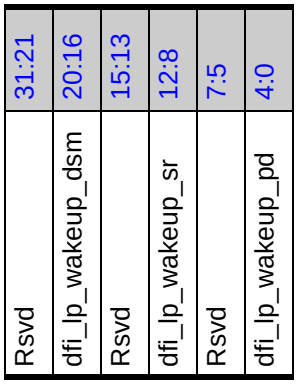
Bits	Name	Memory Access	Description
31:24	dfi_twck_fast_toggle	R/W	Defines the number of clock cycles between the dfi_wck_signal being driven to TOGGLE to when the dfi_wck_signal is driven to FAST_TOGGLE. This timing is only applicable when the WCK transitions from the slow to fast toggle. Otherwise, this timing parameter must be set to 0x0. Unit: WCK cycles **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 4
23:16	dfi_twck_toggle	R/W	Defines the number of clock cycles between the dfi_wck_en signal being enabled to when the dfi_wck_toggle signal is driven to TOGGLE. Unit: WCK cycles **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 4

Bits	Name	Memory Access	Description
15:8	dfi_twck_toggle_cs	R/W	Defines the number of clock cycles between a read or write command to when the dfi_wck_cs signal must be stable. This timing is applicable when the WCK is synchronized for multiple CS's and commands are to different CS's. During WCK synchronization, the CS should be static from the CAS command to the completion of the synchronization sequence. Unit: WCK cycles **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 4
7:0	dfi_twck_toggle_post	R/W	Defines the number of clock cycles after a read or write command data burst completion during which the WCK must remain in the current toggle state. During this time, the dfi_wck_cs signal must also remain stable. Unit: WCK cycles **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 4

1.1.28 DFILPTMG0

- **Description:** DFI Low Power Timing Register 0
- **Size:** 32 bits
- **Offset:** 0x5a0+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.
This register is only present in REGB_FREQ0_CHc, therefore f=0 for the offset calculation



31:21	20:16	15:13	12:8	7:5	4:0
Rsvd	dfi_lp_wakeup_dsm	Rsvd	dfi_lp_wakeup_sr	Rsvd	dfi_lp_wakeup_pd

Table 1-32 Fields for Register: DFILPTMG0

Bits	Name	Memory Access	Description
31:21			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
20:16	dfi_lp_wakeup_dsm	R/W	Indicates the value in DFI clock cycles to drive on dfi_lp_ctrl_wakeup and dfi_lp_data_wakeup signals when Deep Sleep Mode is entered. Determines the DFI's tlp_wakeup time: ■ 0x00 - 1 cycle ■ 0x01 - 2 cycles ■ 0x02 - 4 cycles ■ 0x03 - 8 cycles ■ 0x04 - 16 cycles ■ 0x05 - 32 cycles ■ 0x06 - 64 cycles ■ 0x07 - 128 cycles ■ 0x08 - 256 cycles ■ 0x09 - 512 cycles ■ 0x0A - 1024 cycles ■ 0x0B - 2048 cycles ■ 0x0C - 4096 cycles ■ 0x0D - 8192 cycles ■ 0x0E - 16384 cycles ■ 0x0F - 32768 cycles ■ 0x10 - 65536 cycles ■ 0x11 - 131072 cycles ■ 0x12 - 262144 cycles ■ 0x13 - Unlimited This is only present for designs supporting LPDDR5 devices. Unit: DFI clock cycles. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Static
15:13			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
12:8	dfi_lp_wakeup_sr	R/W	Indicates the value in DFI clock cycles to drive on dfi_lp_ctrl_wakeup and dfi_lp_data_wakeup signals when Self Refresh mode is entered. Determines the DFI's tlp_wakeup time: ■ 0x00 - 1 cycle ■ 0x01 - 2 cycles ■ 0x02 - 4 cycles ■ 0x03 - 8 cycles ■ 0x04 - 16 cycles ■ 0x05 - 32 cycles ■ 0x06 - 64 cycles ■ 0x07 - 128 cycles ■ 0x08 - 256 cycles ■ 0x09 - 512 cycles ■ 0x0A - 1024 cycles ■ 0x0B - 2048 cycles ■ 0x0C - 4096 cycles ■ 0x0D - 8192 cycles ■ 0x0E - 16384 cycles ■ 0x0F - 32768 cycles ■ 0x10 - 65536 cycles ■ 0x11 - 131072 cycles ■ 0x12 - 262144 cycles ■ 0x13 - Unlimited Note: For Synopsys DDR54 PHY, this field must be set greater than 0 (0x01-0x0F is recommended currently). Unit: DFI clock cycles. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:5			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
4:0	dfi_lp_wakeup_pd	R/W	Indicates the value in DFI clock cycles to drive on dfi_lp_ctrl_wakeup and dfi_lp_data_wakeup signals when Power Down mode is entered. Determines the DFI's tlp_wakeup time: ■ 0x00 - 1 cycle ■ 0x01 - 2 cycles ■ 0x02 - 4 cycles ■ 0x03 - 8 cycles ■ 0x04 - 16 cycles ■ 0x05 - 32 cycles ■ 0x06 - 64 cycles ■ 0x07 - 128 cycles ■ 0x08 - 256 cycles ■ 0x09 - 512 cycles ■ 0x0A - 1024 cycles ■ 0x0B - 2048 cycles ■ 0x0C - 4096 cycles ■ 0x0D - 8192 cycles ■ 0x0E - 16384 cycles ■ 0x0F - 32768 cycles ■ 0x10 - 65536 cycles ■ 0x11 - 131072 cycles ■ 0x12 - 262144 cycles ■ 0x13 - Unlimited Note: For Synopsys DDR54 PHY, this field must be set greater than 0 (0x01-0x0F is recommended currently). Unit: DFI clock cycles. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.1.29 DFILPTMG1

- **Description:** DFI Low Power Timing Register 1
- **Size:** 32 bits
- **Offset:** 0x5a4+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.
This register is only present in REGB_FREQ0_CHc, therefore f=0 for the offset calculation

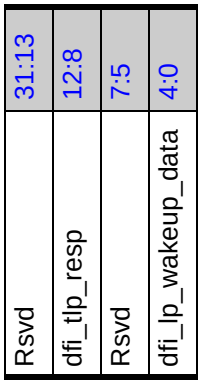


Table 1-33 Fields for Register: DFILPTMG1

Bits	Name	Memory Access	Description
31:13			**Reserved Field:** Yes
12:8	dfi_tlp_resp	R/W	Setting in DFI clock cycles for DFI's tlp_resp time. Same value is used for both Power Down, Self Refresh, Deep Sleep Mode and Maximum Power Saving modes. Refer to PHY databook for recommended values Unit: DFI clock cycles. **Value After Reset:** 0x7 **Exists:** Always **Programming Mode:** Static
7:5			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
4:0	dfi_lp_wakeup_data	R/W	Indicates the value in DFI clock cycles to drive on dfi_lp_data_wakeup signal when data bus is idle. Determines the DFI's tlp_wakeup time: ■ 0x00 - 1 cycle ■ 0x01 - 2 cycles ■ 0x02 - 4 cycles ■ 0x03 - 8 cycles ■ 0x04 - 16 cycles ■ 0x05 - 32 cycles ■ 0x06 - 64 cycles ■ 0x07 - 128 cycles ■ 0x08 - 256 cycles ■ 0x09 - 512 cycles ■ 0x0A - 1024 cycles ■ 0x0B - 2048 cycles ■ 0x0C - 4096 cycles ■ 0x0D - 8192 cycles ■ 0x0E - 16384 cycles ■ 0x0F - 32768 cycles ■ 0x10 - 65536 cycles ■ 0x11 - 131072 cycles ■ 0x12 - 262144 cycles ■ 0x13 - Unlimited Unit: DFI clock cycles. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.1.30 DFIUPDTMG0

- **Description:** DFI Update Timing Register 0
- **Size:** 32 bits
- **Offset:** 0x5a8+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.
This register is only present in REGB_FREQ0_CHc, therefore f=0 for the offset calculation

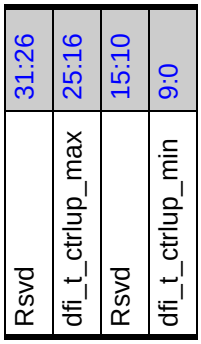


Table 1-34 Fields for Register: DFIUPDTMG0

Bits	Name	Memory Access	Description
31:26			**Reserved Field:** Yes
25:16	dfi_t_ctrlup_max	R/W	Specifies the maximum number of DFI clock cycles that the dfi_ctrlupd_req signal can assert. Lowest value to assign to this variable is 0x40. Unit: DFI clock cycles. **Value After Reset:** 0x40 **Exists:** Always **Programming Mode:** Static
15:10			**Reserved Field:** Yes
9:0	dfi_t_ctrlup_min	R/W	Specifies the minimum number of DFI clock cycles that the dfi_ctrlupd_req signal must be asserted. The DDRCTL expects the PHY to respond within this time. If the PHY does not respond, the DDRCTL will de-assert dfi_ctrlupd_req after dfi_t_ctrlup_min + 2 cycles. Lowest value to assign to this variable is 0x1. Unit: DFI clock cycles. **Value After Reset:** 0x3 **Exists:** Always **Programming Mode:** Static

1.1.31 DFIUPDTMG1

- **Description:** DFI Update Timing Register 1
- **Size:** 32 bits
- **Offset:** 0x5ac+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

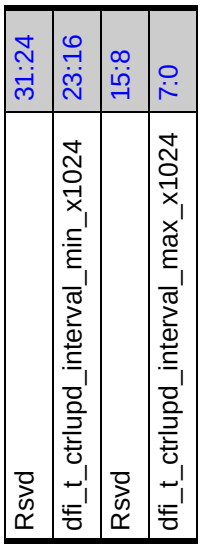


Table 1-35 Fields for Register: DFIUPDTMG1

Bits	Name	Memory Access	Description
31:24			**Reserved Field:** Yes
23:16	dfi_t_ctrlupd_interval_min_x1024	R/W	This is the minimum amount of time between DDRCTL initiated DFI update type0 requests (which is executed whenever the DDRCTL is idle). Set this number higher to reduce the frequency of update requests, which can have a small impact on the latency of the first read request when the DDRCTL is idle. Minimum allowed value for this field is 1. Unit: Multiples of 1024 DFI clock cycles. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Static
15:8			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
7:0	dfi_t_ctrlupd_interval_max_x1024	R/W	This is the maximum amount of time between DDRCTL initiated DFI update type0 requests. This timer resets with each update request; when the timer expires dfi_ctrlupd_req is sent and traffic is blocked until the dfi_ctrlupd_ackx is received. PHY can use this idle time to recalibrate the delay lines to the DLLs. The DFI controller update is also used to reset PHY FIFO pointers in case of data capture errors. Updates are required to maintain calibration over PVT, but frequent updates may impact performance. Minimum allowed value for this field is 1. Note: Value programmed for DFIUPDTMG1.dfi_t_ctrlupd_interval_max_x1024 must be greater than DFIUPDTMG1.dfi_t_ctrlupd_interval_min_x1024. Unit: Multiples of 1024 DFI clock cycles. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Static

1.1.32 DFIMSGTMG0

- **Description:** DFI MC-PHY Message Timing Register 0
- **Size:** 32 bits
- **Offset:** 0x5b0+f*0x100000+c*0x1000
- **Exists:** DDRCTL_DFI_CTRLMSG==1

This register is in block REGB_FREQf_CHc.
This register is only present in REGB_FREQ0_CHc, therefore f=0 for the offset calculation

Table 1-36 Fields for Register: DFIMSGTMG0

Bits	Name	Memory Access	Description
31:8			**Reserved Field:** Yes
7:0	dfi_t_ctrlmsg_resp	R/W	This is the maximum amount in DFI clock cycles between the assertion of the dfi{0/1}_ctrlmsg_req signal to the assertion of the dfi{0/1}_ctrlmsg_ack signal. If the PHY does not acknowledge the request within dfi_t_ctrlmsg_resp cycles, the PHY must not acknowledge the request at all. In this case, the controller should de-assert the corresponding dfi{0/1}_ctrlmsg_req signal. The timing values might vary based on the frequency ratio and user must reprogram if there is any change in the frequency ratio. Refer to PHY databook for recommended values Unit: DFI clock cycles. **Value After Reset:** 0x4 **Exists:** Always **Programming Mode:** Static

1.1.33 DFIUPDTMG2

- **Description:** DFI Update Timing Register 2
- **Size:** 32 bits
- **Offset:** 0x5b4+f*0x100000+c*0x1000
- **Exists:** DDRCTL_PPT2==1

This register is in block REGB_FREQf_CHc.

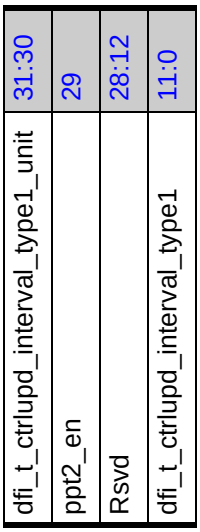


Table 1-37 Fields for Register: DFIUPDTMG2

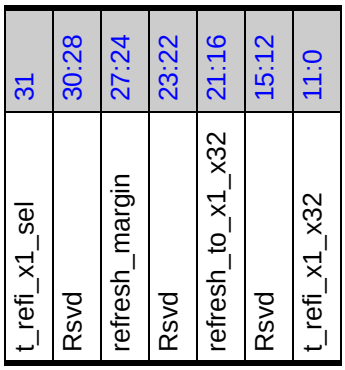
Bits	Name	Memory Access	Description
31:30	dfi_t_ctrlupd_interval_type1_unit	R/W	DFI update type1 (PPT2) interval unit. Specifies the unit for counting interval. The value can be changed while DFIUPDTMG2.ppt2_en=0 and PPT2STAT0.ppt2_burst_busy=0 ■ 0: x32 DFI clock cycles (FOR DEBUG ONLY) ■ 1: x1k DFI clock cycles ■ 2: x16k DFI clock cycles ■ 3: x256k DFI clock cycles **Value After Reset:** 0x3 **Exists:** DDRCTL_PPT2==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
29	ppt2_en	R/W	This register indicates if Normal PPT2 enable is enabled or disabled. ■ 0: Disable Normal PPT2 ■ 1: Enable Normal PPT2 The value can be changed while DDRCTL is in Normal mode **Value After Reset:** 0x0 **Exists:** DDRCTL_PPT2==1 **Programming Mode:** Dynamic
28:12			**Reserved Field:** Yes
11:0	dfi_t_ctrlupd_interval_type1	R/W	This is the amount of time between DDRCTL initiated DFI update type1 (PPT2) requests. The timer resets with each update request. Set this number higher to reduce the frequency of update requests; when the timer expires, dfi_ctrlupd_req is sent and traffic is blocked for 500ns at most until DDRCTL receives dfi_ctrlupd_ack from PHY. PHY can use this idle time to retrain the DQ, DQS and WCK bus. Updates are required to maintain calibration over PVT, but too frequent updates may impact performance. Minimum allowed value for this field is 1. The value can be changed while DFIUPDTMG2.ppt2_en=0 and PPT2STAT0.ppt2_burst_busy=0 Unit: determined by DFIUPDTMG2.dfi_t_ctrlupd_interval_type1_unit Please refer to "Retraining interval" at the start of "PPT2 - Enhanced incremental periodic phase training" chapter for details on how to program this register field. **Value After Reset:** 0x12c **Exists:** DDRCTL_PPT2==1 **Programming Mode:** Dynamic

1.1.34 RFSHSET1TMG0

- **Description:** Refresh Timing Register 0 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x600+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.



31	30:28	27:24	23:22	21:16	15:12	11:0
t_refi_x1_sel	Rsvd	refresh_margin	Rsvd	refresh_to_x1_x32	Rsvd	t_refi_x1_x32

Table 1-38 Fields for Register: RFSHSET1TMG0

Bits	Name	Memory Access	Description
31	t_refi_x1_sel	R/W	Specifies whether RFSHSET1TMG0.t_refi_x1_x32 and RFSHSET1TMG0.refresh_to_x1_x32 register values are x1 or x32. ■ 0 - x32 register values are used, ■ 1 - x1 register values are used. This applies only when per-bank refresh is enabled (RFSHMOD0.per_bank_refresh=1); if per-bank refresh is not enabled, the x32 register values are used and this register field is ignored. This register field does not exist for configurations which do not support LPDDR4/5. For such configurations, the value of this register field can be assumed to be 0, so that RFSHSET1TMG0.t_refi_x1_x32 and RFSHSET1TMG0.refresh_to_x1_x32 are interpreted as x32 register fields **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic - Refresh Related
30:28			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
27:24	refresh_margin	R/W	Threshold value in number of DRAM clock cycles before the critical refresh or page timer expires. A critical refresh is to be issued before this threshold is reached. It is recommended that this not be changed from the default value, currently shown as 0x2. It must always be less than internally used t_refi/32. Note that internally used t_refi is equal to RFSHSET1TMG0.t_refi_x1_x32 * 32 if RFSHSET1TMG0.t_refi_x1_sel = 0. If RFSHSET1TMG0.t_refi_x1_sel = 1, internally used t_refi is equal to RFSHSET1TMG0.t_refi_x1_x32. Note that, internally used t_refi may be divided by four if derating or TCR is enabled. Unit: Multiples of 32 DRAM clock cycles. **Value After Reset:** 0x2 **Exists:** Always **Programming Mode:** Dynamic - Refresh Related
23:22			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
21:16	refresh_to_x1_x32	R/W	If the refresh timer has expired at least once (i.e. >tREFI period elapses, and there are postponed refreshes), then a speculative refresh may be performed. A speculative refresh is a refresh performed at a time when refresh would be useful. When there are no transactions pending in the CAM for a period of time determined by this RFSHSET1TMG0.refresh_to_x1_x32 and the refresh timer has expired at least once since the last refresh, then a speculative refresh is performed. Speculative refreshes continues successively until there are no refreshes pending or until new reads or writes are issued to the DDRCTL. This is also used for non speculative refresh when LPDDR per-bank refresh (REFpb) or DDR5 same-bank refresh (REFsb) is enabled. The controller observes the period of time determined by this for each bank, and a priority of bank address is determined. For non-DDR5, this should be programmed to tREFI based value in controller's current refresh mode. For DDR5, this should be always programmed to tREFI1 based value even in FGR mode. The controller calculates this according to current refresh mode. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. FOR PERFORMANCE ONLY. Unit: DRAM clock cycles or multiples of 32 DRAM clock cycles, depending on RFSHSET1TMG0.t_refi_x1_sel. **Value After Reset:** 0x10 **Exists:** Always **Programming Mode:** Dynamic - Refresh Related
15:12			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
11:0	t_refi_x1_x32	R/W	Average time interval between refreshes per rank (Specification: 7.8us for DDR4, 3.9us for DDR5. See JEDEC specification for LPDDR4/LPDDR5). set this register to RoundDown(tREFI/tCK) if RFSHSET1TMG0.t_refi_x1_sel = 0, divide the above result by 32 and round down. For LPDDR controller: ■ if using all-bank refreshes (RFSHMOD0.per_bank_refresh = 0), use tREFIab in the above calculations ■ if using per-bank refreshes (RFSHMOD0.per_bank_refresh = 1), use tREFIpb in the above calculations For DDR controller, tREFI value is different depending on FGR mode. ■ In DDR4 mode, if using FGR 1x mode (RFSHMOD1.fgr_mode = 000), use tREFI1 in the above calculations ■ In DDR4 mode, if using FGR 2x mode (RFSHMOD1.fgr_mode = 001), use tREFI2 in the above calculations ■ In DDR4 mode, if using FGR 4x mode (RFSHMOD1.fgr_mode = 010), use tREFI4 in the above calculations ■ In DDR5 mode, always use tREFI1 in the above calculations Note that: ■ RFSHSET1TMG0.t_refi_x1_x32 must be greater than 0x1. ■ if RFSHSET1TMG0.t_refi_x1_sel == 1, RFSHSET1TMG0.t_refi_x1_x32 must be greater than RFSHSET1TMG1.t_rfc_min ■ if RFSHSET1TMG0.t_refi_x1_sel == 0, RFSHSET1TMG0.t_refi_x1_x32 * 32 must be greater than RFSHSET1TMG1.t_rfc_min ■ In non-DDR4 or DDR4 Fixed 1x mode: RFSHSET1TMG0.t_refi_x1_x32 must be less than or equal to 0xFFE. ■ In DDR4 Fixed 2x mode: RFSHSET1TMG0.t_refi_x1_x32 must be less than or equal to 0x7FF. ■ In DDR4 Fixed 4x mode: RFSHSET1TMG0.t_refi_x1_x32 must be less than or equal to 0x3FF. Unit: DRAM clock cycles or multiples of 32 DRAM clock cycles, depending on RFSHSET1TMG0.t_refi_x1_sel. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x62 **Exists:** Always **Programming Mode:** Dynamic - Refresh Related

1.1.35 RFSHSET1TMG1

- **Description:** Refresh Timing Register 1 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x604+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

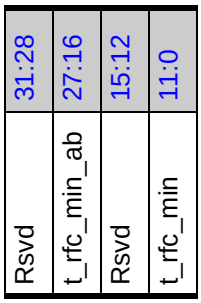


Table 1-39 Fields for Register: RFSHSET1TMG1

Bits	Name	Memory Access	Description
31:28			**Reserved Field:** Yes
27:16	t_rfc_min_ab	R/W	tRFCab: Minimum time from refresh to refresh or activate, for all-bank refreshes. When RFSHMOD0.auto_refab_en > 0 and RFSHMOD0.per_bank_refresh == 1, the controller will use this value when switching automatically from per-bank refresh to all-bank refresh if the derated refresh period is too small. t_rfc_min_ab must be set to RoundUp(tRFCab/tCK). Must be set for LPDDR5 2Gb, 6Gb, and 8Gb. Must be set to 0 for other device densities. Unit: DRAM clock cycles. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic - Refresh Related
15:12			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
11:0	t_rfc_min	R/W	tRFC (min): Minimum time from refresh to refresh or activate. t_rfc_min must be set to RoundUp(tRFCmin/tCK). In LPDDR controller: ■ if using all-bank refreshes, the tRFCmin value in the above equations is equal to tRFCab ■ if using per-bank refreshes, the tRFCmin value in the above equations is equal to tRFCpb In DDR4/DDR5 mode, the tRFCmin value in the above equations is different depending on the refresh mode (fixed 1X,2X,4X) and the device density. The user must program the appropriate value from the spec based on the 'fgr_mode' and the device density that is used. Unit: DRAM clock cycles. **Value After Reset:** 0x8c **Exists:** Always **Programming Mode:** Dynamic - Refresh Related

1.1.36 RFSHSET1TMG2

- **Description:** Refresh Timing Register 2 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x608+f*0x100000+c*0x1000
- **Exists:** MEMC_LPDDR4_OR_UMCTL2_CID_EN==1

This register is in block REGB_FREQf_CHc.

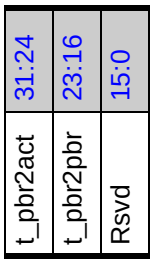


Table 1-40 Fields for Register: RFSHSET1TMG2

Bits	Name	Memory Access	Description
31:24	t_pbr2act	R/W	Time from REFpb to activate command to different bank than REFpb. LPDDR5: tpbr2act **Value After Reset:** 0x8c **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic - Refresh Related
23:16	t_pbr2pbr	R/W	LPDDR4/LPDDR5: tpbR2pbR Per-bank Refresh to Per-bank refresh different bank Time. Program this to RoundUp(tpbR2pbR/tCK). The tpbR2pbR value in the above equations is different depending on the device density. The user must program the appropriate value from the spec. Register is valid only in per-bank refresh mode (RFSHMOD0.per_bank_refresh == 1). **Value After Reset:** 0x8c **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic - Refresh Related
15:0			**Reserved Field:** Yes

1.1.37 RFSHSET1TMG3

- **Description:** Refresh Timing Register 3 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x60c+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.
Note: This register is only applicable for designs supporting DDR5 or LPDDR54 SDRAM memories

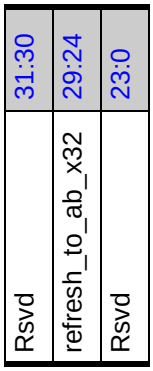


Table 1-41 Fields for Register: RFSHSET1TMG3

Bits	Name	Memory Access	Description
31:30			**Reserved Field:** Yes
29:24	refresh_to_ab_x32	R/W	When the DDRCTL switches automatically from per-bank to all-bank refresh (if enabled by RFSHMOD0.auto_refab_en), it will use this register to determine when to perform speculative all-bank refreshes. If the refresh timer (tRFCnom, also known as tREFI) has expired at least once, then a speculative refresh may be performed. A speculative refresh is a refresh performed at a time when refresh would be useful. When the SDRAM bus is idle for a period of time determined by this RFSHSET1TMG3.refresh_to_ab_x32 and the refresh timer has expired at least once since the last refresh, then a speculative refresh is performed. Speculative refreshes continues successively until there are no refreshes pending or until new reads or writes are issued to the DDRCTL. FOR PERFORMANCE ONLY. Unit: Multiples of 32 DRAM clock cycles Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x10 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic - Refresh Related

Bits	Name	Memory Access	Description
23:0			**Reserved Field:** Yes

1.1.38 RFSHSET1TMG4

- **Description:** Refresh Timing Register 4 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x610+f*0x100000+c*0x1000
- **Exists:** MEMC_NUM_RANKS>1

This register is in block REGB_FREQf_CHc.

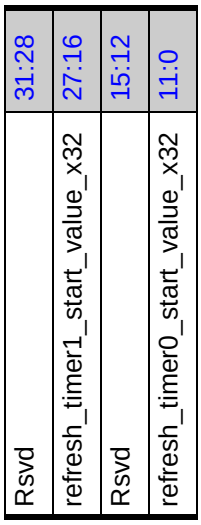


Table 1-42 Fields for Register: RFSHSET1TMG4

Bits	Name	Memory Access	Description
31:28			**Reserved Field:** Yes
27:16	refresh_timer1_start_value_x32	R/W	Refresh timer start for rank 1 (only present in multi-rank configurations). This is useful in staggering the refreshes to multiple ranks to help traffic to proceed. This is explained in Refresh Controls section of architecture chapter. FOR PERFORMANCE ONLY. Unit: Multiples of 32 DRAM clock cycles. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x0 **Exists:** MEMC_NUM_RANKS>1 **Programming Mode:** Dynamic - Refresh Related
15:12			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
11:0	refresh_timer0_start_value_x32	R/W	Refresh timer start for rank 0 (only present in multi-rank configurations). This is useful in staggering the refreshes to multiple ranks to help traffic to proceed. This is explained in Refresh Controls section of architecture chapter. FOR PERFORMANCE ONLY. Unit: Multiples of 32 DRAM clock cycles. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x0 **Exists:** MEMC_NUM_RANKS>1 **Programming Mode:** Dynamic - Refresh Related

1.1.39 RFSHSET1TMG5

- **Description:** Refresh Timing Register 5 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x614+f*0x100000+c*0x1000
- **Exists:** MEMC_NUM_RANKS>2

This register is in block REGB_FREQf_CHc.

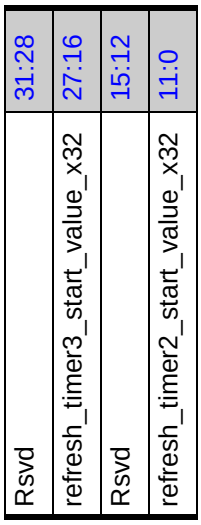


Table 1-43 Fields for Register: RFSHSET1TMG5

Bits	Name	Memory Access	Description
31:28			**Reserved Field:** Yes
27:16	refresh_timer3_start_value_x32	R/W	Refresh timer start for rank 3 (only present in 4-rank configurations). This is useful in staggering the refreshes to multiple ranks to help traffic to proceed. This is explained in Refresh Controls section of architecture chapter. FOR PERFORMANCE ONLY. Unit: Multiples of 32 DRAM clock cycles. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x0 **Exists:** MEMC_NUM_RANKS>2 **Programming Mode:** Dynamic - Refresh Related
15:12			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
11:0	refresh_timer2_start_value_x32	R/W	Refresh timer start for rank 2 (only present in 4-rank configurations). This is useful in staggering the refreshes to multiple ranks to help traffic to proceed. This is explained in Refresh Controls section of architecture chapter. FOR PERFORMANCE ONLY. Unit: Multiples of 32 DRAM clock cycles. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x0 **Exists:** MEMC_NUM_RANKS>2 **Programming Mode:** Dynamic - Refresh Related

1.1.40 RFMSET1TMG0

- **Description:** RFM Timing Register 0 belonging to Timing Set 1
- **Size:** 32 bits
- **Offset:** 0x650+f*0x100000+c*0x1000
- **Exists:** DDRCTL_LPDDR_RFM==1

This register is in block REGB_FREQf_CHc.

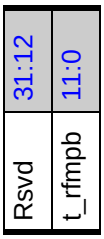


Table 1-44 Fields for Register: RFMSET1TMG0

Bits	Name	Memory Access	Description
31:12			**Reserved Field:** Yes
11:0	t_rfmpb	R/W	tRFMpb: Refresh Management Cycle (per-bank). Unit: DRAM clock cycles. **Value After Reset:** 0x8c **Exists:** Always **Programming Mode:** Static

1.1.41 ZQSET1TMG0

- **Description:** ZQ Timing Register 0 belonging to DRAM ZQ timing set 1
- **Size:** 32 bits
- **Offset:** 0x800+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

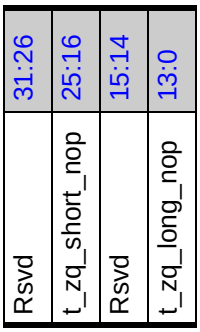


Table 1-45 Fields for Register: ZQSET1TMG0

Bits	Name	Memory Access	Description
31:26			**Reserved Field:** Yes
25:16	t_zq_short_nop	R/W	tZQCS for DD4, tZQLAT for DDR5/LPDDR4/LPDDR5: Number of DRAM clock cycles of NOP required after a ZQCS (ZQ calibration short)/MPC(ZQ Latch) command is issued to SDRAM. Unit: DRAM clock cycles. **Value After Reset:** 0x40 **Exists:** Always **Programming Mode:** Static
15:14			**Reserved Field:** Yes
13:0	t_zq_long_nop	R/W	tZQoper for DDR4, tZQCAL for DDR5/LPDDR4/LPDDR5: Number of DRAM clock cycles of NOP required after a ZQCL (ZQ calibration long)/MPC(ZQ Start) command is issued to SDRAM. If using LPDDR5, this register needs to be programmed to tZQCAL + 10 cycles. Unit: DRAM clock cycles. **Value After Reset:** 0x200 **Exists:** Always **Programming Mode:** Static

1.1.42 ZQSET1TMG1

- **Description:** ZQ Timing Register 1 belonging to DRAM ZQ timing set 1
- **Size:** 32 bits
- **Offset:** 0x804+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

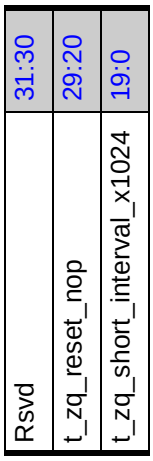


Table 1-46 Fields for Register: ZQSET1TMG1

Bits	Name	Memory Access	Description
31:30			**Reserved Field:** Yes
29:20	t_zq_reset_nop	R/W	tZQReset: Number of DRAM clock cycles of NOP required after a ZQReset (ZQ calibration Reset) command is issued to SDRAM. Unit: DRAM clock cycles. **Value After Reset:** 0x20 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Static
19:0	t_zq_short_interval_x1024	R/W	Average interval to wait between automatically issuing ZQCS (ZQ calibration short)/MPC(ZQ calibration) commands to DDR4/LPDDR4 devices. Meaningless, if ZQCTL0.dis_auto_zq=1. Unit: Multiples of 1024 DRAM clock cycles. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x100 **Exists:** Always **Programming Mode:** Static

1.1.43 DQSOSCCTL0

- **Description:** DQS/WCK Oscillator Control Register 0
- **Size:** 32 bits
- **Offset:** 0xa80+f*0x100000+c*0x1000
- **Exists:** LPDDR45_DQSOSC_EN==1

This register is in block REGB_FREQf_CHc.

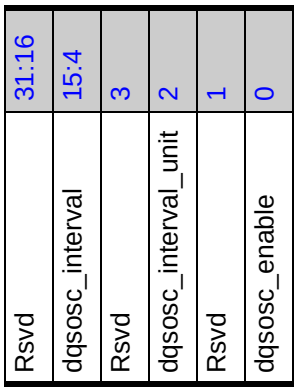


Table 1-47 Fields for Register: DQSOSCCTL0

Bits	Name	Memory Access	Description
31:16			**Reserved Field:** Yes
15:4	dqsosc_interval	R/W	DQS Oscillator interval, specifies the time between two DQS oscillator sequences. Minimum programmable value is 1. The value can be changed while DQSOSCCTL0.dqsosc_enable=0 Unit: DFI clock cycles **Value After Reset:** 0x7 **Exists:** LPDDR45_DQSOSC_EN==1 **Programming Mode:** Dynamic
3			**Reserved Field:** Yes
2	dqsosc_interval_unit	R/W	DQS/WCK Oscillator Interval unit. Specifies the unit for counting DQS oscillator interval.The value can be changed while DQSOSCCTL0.dqsosc_enable=0 1: x2K DFI clock cycles 0: x32K DFI clock cycles **Value After Reset:** 0x0 **Exists:** LPDDR45_DQSOSC_EN==1 **Programming Mode:** Dynamic
1			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
0	dqsosc_enable	R/W	DQS/WCK Oscillator Enable 1: Enable DQS Oscillator 0: Disable DQS Oscillator **Value After Reset:** 0x0 **Exists:** LPDDR45_DQSOSC_EN==1 **Programming Mode:** Dynamic

1.1.44 DERATEINT

- **Description:** Temperature Derate Interval Register
- **Size:** 32 bits
- **Offset:** 0xb00+f*0x100000+c*0x1000
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_FREQf_CHc.

Table 1-48 Fields for Register: DERATEINT

Bits	Name	Memory Access	Description
31:0	mr4_read_interval	R/W	Interval between two MR4 reads, used to derate the timing parameters. This register must not be set to zero. Unit: DRAM clock cycles. **Value After Reset:** 0x800000 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Static

1.1.45 DERATEVAL0

- **Description:** Temperature Derate Timing Register 0
- **Size:** 32 bits
- **Offset:** 0xb04+f*0x100000+c*0x1000
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_FREQf_CHc.

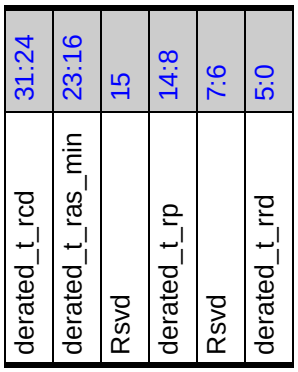


Table 1-49 Fields for Register: DERATEVAL0

Bits	Name	Memory Access	Description
31:24	derated_t_rcd	R/W	Derated value for tRCD. For LPDDR4, the required period with derating is tRCD + 1.875ns For LPDDR5, the required period with derating is tRCD + TBD ns This timing parameter must be rounded up to the next integer value. Unit:DRAM clock cycles. **Value After Reset:** 0x5 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

Bits	Name	Memory Access	Description
23:16	derated_t_ras_min	R/W	Derated value for tRAS. For LPDDR4, the required period with derating is tRAS + 1.875ns For LPDDR5, the required period with derating is tRAS + TBD ns This timing parameter must be rounded up to the next integer value. Unit:DRAM clock cycles. **Value After Reset:** 0xf **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
15			**Reserved Field:** Yes
14:8	derated_t_rp	R/W	Derated value for tRP. For LPDDR4, the required period with derating is tRP + 1.875ns For LPDDR5, the required period with derating is tRP + TBD ns This timing parameter must be rounded up to the next integer value. Unit:DRAM clock cycles. **Value After Reset:** 0x5 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4
7:6			**Reserved Field:** Yes
5:0	derated_t_rrd	R/W	Derated value for tRRD. For LPDDR4, the required period with derating is tRRD + 1.875ns For LPDDR5, the required period with derating is tRRD + TBD ns This timing parameter must be rounded up to the next integer value. Unit:DRAM clock cycles. **Value After Reset:** 0x4 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

1.1.46 DERATEVAL1

- **Description:** Temperature Derate Timing Register 1
- **Size:** 32 bits
- **Offset:** 0xb08+f*0x100000+c*0x1000
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_FREQf_CHc.

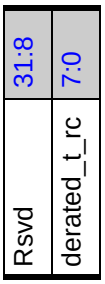


Table 1-50 Fields for Register: DERATEVAL1

Bits	Name	Memory Access	Description
31:8			**Reserved Field:** Yes
7:0	derated_t_rc	R/W	Derated value for tRC. For LPDDR4, the required period with derating is tRC + 3.75ns For LPDDR5, the required period with derating is tRC + TBD ns This timing parameter must be rounded up to the next integer value. Unit:DRAM clock cycles. **Value After Reset:** 0x14 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

1.1.47 HWLPTMG0

- **Description:** Hardware Low Power Control Register
- **Size:** 32 bits
- **Offset:** 0xb80+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

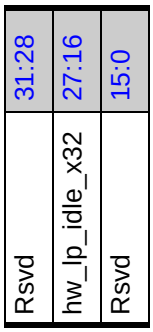


Table 1-51 Fields for Register: HWLPTMG0

Bits	Name	Memory Access	Description
31:28			**Reserved Field:** Yes
27:16	hw_lp_idle_x32	R/W	Hardware idle period. The cactive_ddrc output is driven low if the DDRC command channel is idle for hw_lp_idle * 32 cycles if not in INIT or DPD/MPSM operating_mode. The DDRC command channel is considered idle when there are no HIF commands outstanding. The hardware idle function is disabled when hw_lp_idle_x32=0. FOR PERFORMANCE ONLY. Unit: Multiples of 32 DRAM clock cycles. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Static
15:0			**Reserved Field:** Yes

1.1.48 SCHEDTMG0

- **Description:** Scheduler Control Register
- **Size:** 32 bits
- **Offset:** 0xc00+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

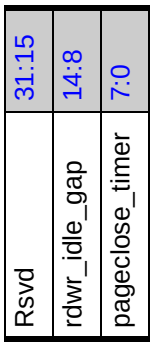


Table 1-52 Fields for Register: SCHEDTMG0

Bits	Name	Memory Access	Description
31:15			**Reserved Field:** Yes
14:8	rdwr_idle_gap	R/W	When the preferred transaction store is empty for these many clock cycles, switch to the alternate transaction store if it is non-empty. The read transaction store (both high and low priority) is the default preferred transaction store and the write transaction store is the alternative store. When prefer write over read is set this is reversed. 0x0 is a legal value for this register. When set to 0x0, the transaction store switching will happen immediately when the switching conditions become true. FOR PERFORMANCE ONLY. Unit: DRAM clock cycles. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

Bits	Name	Memory Access	Description
7:0	pageclose_timer	R/W	This field works in conjunction with SCHED.pageclose. It only has meaning if SCHED.pageclose==1. If SCHED.pageclose==1 and pageclose_timer==0, then an auto-precharge may be scheduled for last read or write command in the CAM with a bank and page hit. Note, sometimes an explicit precharge is scheduled instead of the auto-precharge. See SCHED.pageclose for details of when this may happen. If SCHED.pageclose==1 and pageclose_timer>0, then an auto-precharge is not scheduled for last read or write command in the CAM with a bank and page hit. Instead, a timer is started, with pageclose_timer as the initial value. There is a timer on a per bank basis. The timer decrements unless the next read or write in the CAM to a bank is a page hit. It gets reset to pageclose_timer value if the next read or write in the CAM to a bank is a page hit. Once the timer has reached zero, an explicit precharge will be attempted to be scheduled. Unit: DRAM clock cycles. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.1.49 PERFHPR1

- **Description:** High Priority Read CAM Register 1
- **Size:** 32 bits
- **Offset:** 0xc80+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

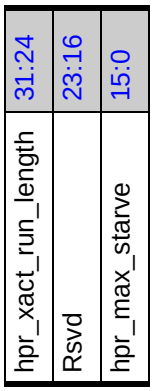


Table 1-53 Fields for Register: PERFHPR1

Bits	Name	Memory Access	Description
31:24	hpr_xact_run_length	R/W	Number of transactions that are serviced once the HPR queue goes critical is the smaller of: ■ This number ■ Number of transactions available Unit: Transaction. FOR PERFORMANCE ONLY. **Value After Reset:** 0xf **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
23:16			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
15:0	hpr_max_starve	R/W	Number of DRAM clocks that the HPR queue can be starved before it goes critical. The minimum valid functional value for this register is 0x1. Programming it to 0x0 will disable the starvation functionality; during normal operation, this function must not be disabled as it will cause excessive latencies. FOR PERFORMANCE ONLY. Unit: DRAM clock cycles. **Value After Reset:** 0x1 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3

1.1.50 PERFLPR1

- **Description:** Low Priority Read CAM Register 1
- **Size:** 32 bits
- **Offset:** 0xc84+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

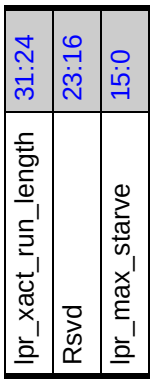


Table 1-54 Fields for Register: PERFLPR1

Bits	Name	Memory Access	Description
31:24	lpr_xact_run_length	R/W	Number of transactions that are serviced once the LPR queue goes critical is the smaller of: ■ This number ■ Number of transactions available Unit: Transaction. FOR PERFORMANCE ONLY. **Value After Reset:** 0xf **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
23:16			**Reserved Field:** Yes
15:0	lpr_max_starve	R/W	Number of DRAM clocks that the LPR queue can be starved before it goes critical. The minimum valid functional value for this register is 0x1. Programming it to 0x0 will disable the starvation functionality; during normal operation, this function must not be disabled as it will cause excessive latencies. FOR PERFORMANCE ONLY. Unit: DRAM clock cycles. **Value After Reset:** 0x7f **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3

1.1.51 PERFWR1

- **Description:** Write CAM Register 1
- **Size:** 32 bits
- **Offset:** 0xc88+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

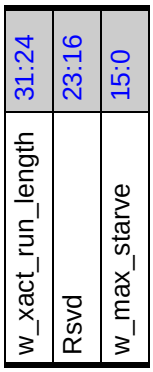


Table 1-55 Fields for Register: PERFWR1

Bits	Name	Memory Access	Description
31:24	w_xact_run_length	R/W	Number of transactions that are serviced once the WR queue goes critical is the smaller of: ■ (a) This number ■ (b) Number of transactions available. Unit: Transaction. FOR PERFORMANCE ONLY. **Value After Reset:** 0xf **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
23:16			**Reserved Field:** Yes
15:0	w_max_starve	R/W	Number of DRAM clocks that the WR queue can be starved before it goes critical. The minimum valid functional value for this register is 0x1. Programming it to 0x0 will disable the starvation functionality; during normal operation, this function must not be disabled as it will cause excessive latencies. FOR PERFORMANCE ONLY. Unit: DRAM clock cycles. **Value After Reset:** 0x7f **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3

1.1.52 TMGCFG

- **Description:** Timing Configuration Register
- **Size:** 32 bits
- **Offset:** 0xd00+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.
This register is only present in REGB_FREQf_CH0, therefore c=0 for the offset calculation

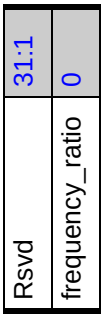


Table 1-56 Fields for Register: TMGCFG

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	frequency_ratio	R/W	Selects the Frequency Ratio For DDR4/DDR5/LPDDR4: ■ 0: 1:2 Mode ■ 1: 1:4 Mode For LPDDR5: ■ 0: 1:1:2 Mode ■ 1: 1:1:4 Mode **Value After Reset:** 0x0 **Exists:** MEMC_PROG_FREQ_RATIO==1 **Programming Mode:** Quasi-dynamic Group 2

1.1.53 RANKTMG0

- **Description:** Rank Control Timing 0
- **Size:** 32 bits
- **Offset:** 0xd04+f*0x100000+c*0x1000
- **Exists:** UMCTL2_NUM_LRANKS_TOTAL>1

This register is in block REGB_FREQf_CHc.

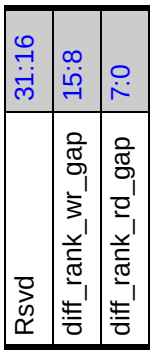


Table 1-57 Fields for Register: RANKTMG0

Bits	Name	Memory Access	Description
31:16			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
15:8	diff_rank_wr_gap	R/W	Only present for multi-rank configurations. Indicates the number of clocks of gap in data responses when performing consecutive writes to different ranks. This is used to switch the delays in the PHY to match the rank requirements. This value must consider both PHY requirement and ODT requirement. ■ PHY requirement: tphy_wrcsgap (see PHY databook for value of tphy_wrcsgap) If CRC feature is enabled, must be increased by 1. If write preamble is set to 2tCK(DDR4 only), must be increased by 1. Write preamble is always set to 2tCK for LPDDR4, refer to PHY databook to see if this is already factored into tphy_wrcsgap value or if it needs to be increased by 1. If write postamble is set to 1.5tCK(LPDDR4 only), must be increased by 1. ■ ODT requirement: The value programmed in this register takes care of the ODT switch off timing requirement when switching ranks during writes. For LPDDR4, with DQ ODT enabled, diff_rank_wr_gap must be a minimum of ODTLoff - ODTLon - BL/2 + 1 For other cases, diff_rank_wr_gap must be a minimum of ODTCFG.wr_odt_hold - BL/2 Program this to the larger of PHY requirement or ODT requirement. After PHY has completed training the value programmed may need to be increased. Refer to relevant PHY documentation. Note that, if using DDR4-LRDIMM, refer to TWRWR timing requirements in JEDEC DDR4 Data Buffer (DDR4DB01) Specification. For LPDDR5, Please set to "JEDEC formula + tphy_wckcsgap + board delay" Please see PHY databook for the value of tphy_wckcsgap Unit: DRAM clock cycles. **Value After Reset:** 0x6 **Exists:** MEMC_NUM_RANKS>1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2

Bits	Name	Memory Access	Description
7:0	diff_rank_rd_gap	R/W	Only present for multi-rank configurations. Indicates the number of clocks of gap in data responses when performing consecutive reads to different ranks. This is used to switch the delays in the PHY to match the rank requirements. This value must consider both PHY requirement and ODT requirement. ■ PHY requirement: tphy_rdcsgap (see PHY databook for value of tphy_rdcsgap) If read preamble is set to 2tCK(DDR4 only), must be increased by 1. If read postamble is set to 1.5tCK(LPDDR4 only), must be increased by 1. ■ ODT requirement: The value programmed in this register takes care of the ODT switch off timing requirement when switching ranks during reads: diff_rank_rd_gap must be a minimum of ODTCFG.rd_odt_hold - BL/2 Program this to the larger of PHY requirement or ODT requirement. After PHY has completed training the value programmed may need to be increased. Refer to relevant PHY documentation. Note that, if using DDR4-LRDIMM, refer to TRDRD timing requirements in JEDEC DDR4 Data Buffer (DDR4DB01) Specification. For LPDDR5, Please set to "JEDEC formula + tphy_wckcsgap + board delay" Please see PHY databook for the value of tphy_wckcsgap Unit: DRAM clock cycles. **Value After Reset:** 0x6 **Exists:** MEMC_NUM_RANKS>1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2

1.1.54 RANKTMG1

- **Description:** Rank Timing Register 1
- **Size:** 32 bits
- **Offset:** 0xd08+f*0x100000+c*0x1000
- **Exists:** MEMC_NUM_RANKS>1

This register is in block REGB_FREQf_CHc.

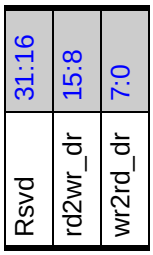


Table 1-58 Fields for Register: RANKTMG1

Bits	Name	Memory Access	Description
31:16			**Reserved Field:** Yes
15:8	rd2wr_dr	R/W	Minimum time from read command to write command for different rank. Includes time for bus turnaround, recovery times, and all per-bank, per-rank, and global constraints. The value must be larger than or equal to the value of DRAMSET1TMG2.rd2wr. For LPDDR5, Please set to "JEDEC formula + tphy_wckcsgap + board delay" Please see PHY databook for the value of tphy_wckcsgap Unit: DRAM clock cycles **Value After Reset:** 0xf **Exists:** MEMC_NUM_RANKS>1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 2, Group 4

Bits	Name	Memory Access	Description
7:0	wr2rd_dr	R/W	Minimum time from write command to read command for different rank. Includes time for bus turnaround, recovery times, and all per-bank, per-rank, and global constraints. For LPDDR5, Please set to "JEDEC formula + tphy_wckcsgap + board delay" Please see PHY databook for the value of tphy_wckcsgap Unit: DRAM clock cycles **Value After Reset:** 0xf **Exists:** MEMC_NUM_RANKS>1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1, Group 2, Group 4

1.1.55 PWRTMG

- **Description:** Low Power Timing Register
- **Size:** 32 bits
- **Offset:** 0xd0c+f*0x100000+c*0x1000
- **Exists:** Always

This register is in block REGB_FREQf_CHc.

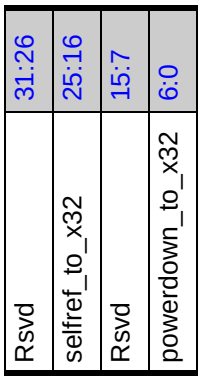


Table 1-59 Fields for Register: PWRTMG

Bits	Name	Memory Access	Description
31:26			**Reserved Field:** Yes
25:16	selfref_to_x32	R/W	After this many clocks of the DDRC command channel being idle the DDRCTL automatically puts the SDRAM into Self Refresh. The DDRC command channel is considered idle when there are no HIF commands outstanding. This must be enabled in the PWRCTL.selfref_en. selfref_to_x32=0 is an illegal value. FOR PERFORMANCE ONLY. Unit: Multiples of 32 DRAM clock cycles. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x40 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 4
15:7			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
6:0	powerdown_to_x32	R/W	After this many clocks of the DDRC command channel being idle the DDRCTL automatically puts the SDRAM into power-down. The DDRC command channel is considered idle when there are no HIF commands outstanding. This must be enabled in the PWRCTL.powerdown_en. This value should be greater than 2 when the controller is in 1:2 mode and 4 when the controller is in 1:4 mode. FOR PERFORMANCE ONLY. Unit: Multiples of 32 DRAM clock cycles. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x10 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 4

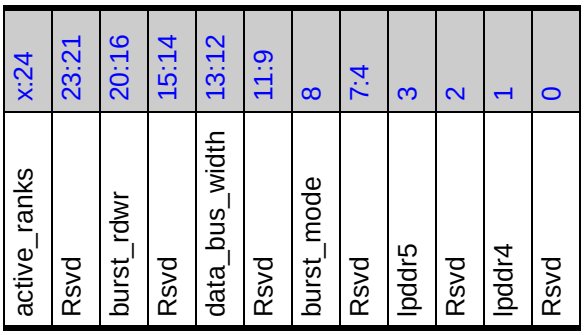
1.2 REGB_DDRC_CH0 Registers

This register block contains registers related to the control of functionality and the configuration of the DDRC controller. Registers shared by both channels are only present in REGB_DDRC_CH0.

1.2.1 MSTR0

- **Description:** Master Register0
- **Size:** 32 bits
- **Offset:** 0x10000
- **Exists:** Always

This register is in block REGB_DDRC_CH0.



x:24	23:21	20:16	15:14	13:12	11:9	8	7:4	3	2	1	0
active_ranks	Rsvd	burst_rdwr	Rsvd	data_bus_width	Rsvd	burst_mode	Rsvd	lpddr5	Rsvd	lpddr4	Rsvd

Table 1-60 Fields for Register: MSTR0

Bits	Name	Memory Access	Description
x:24	active_ranks	R/W	Only present for multi-rank configurations. Each bit represents one rank. For two-rank configurations, only bits[25:24] are present. ■ 1 - populated ■ 0 - unpopulated LSB is the lowest rank number. For 2 ranks following combinations are legal: ■ 01 - One rank ■ 11 - Two ranks ■ Others - Reserved. For 4 ranks following combinations are legal: ■ 0001 - One rank ■ 0011 - Two ranks ■ 0101 - Two ranks are populated in Rank0 and Rank2 (DDR5 Only). ■ 1111 - Four ranks Note: the four rank populated config 4'b0101 can only be supported with heterogeneous rank support enable. **Value After Reset:** "(MEMC_NUM_RANKS==4) ? 0xF :((MEMC_NUM_RANKS==2) ? 0x3 : 0x1)" **Exists:** MEMC_NUM_RANKS>1 **Programming Mode:** Static **Range Variable[x]:** "(MEMC_NUM_RANKS==2) ? 2 : 4" + 23

Bits	Name	Memory Access	Description
23:21			**Reserved Field:** Yes
20:16	burst_rdwr	R/W	SDRAM burst length used: ■ 00100 - Burst length of 8 ■ 01000 - Burst length of 16 All other values are reserved. This controls the burst size used to access the SDRAM. This must match the burst length mode register setting in the SDRAM. For DDR4, this must be set to 0x00100 (BL8). For LPDDR4/LPDDR5/DDR5, this must be set to 0x01000 (BL16). **Value After Reset:** 0x4 **Exists:** Always **Programming Mode:** Static
15:14			**Reserved Field:** Yes
13:12	data_bus_width	R/W	Selects proportion of DQ bus width that is used by the SDRAM ■ 00 - Full DQ bus width to SDRAM ■ 01 - Half DQ bus width to SDRAM ■ 10 - Quarter DQ bus width to SDRAM ■ 11 - Reserved. Note that half bus width mode is only supported when the SDRAM bus width (DQ bus width) is a multiple of 16, and quarter bus width mode is only supported when the SDRAM bus width (DQ bus width) is a multiple of 32 and the configuration parameter MEMC_QBUS_SUPPORT is set. Bus width refers to DQ bus width (excluding any ECC width). However, MEMC_DRAM_DATA_WIDTH = 72 represents 64-bit DQ bus width plus 8-bit ECC and hence supports half and quarter bus width mode. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
11:9			**Reserved Field:** Yes
8	burst_mode	R/W	Indicates burst mode. ■ 0 - Sequential burst mode ■ 1 - Interleaved burst mode For LPDDR4, this must be set to 0 (sequential mode). **Value After Reset:** 0x0 **Exists:** UMCTL2_INCL_ARB==0 **Programming Mode:** Static

Bits	Name	Memory Access	Description
7:4			**Reserved Field:** Yes
3	lpddr5	R/W	Select LPDDR5 SDRAM ■ 1 - LPDDR5 SDRAM device in use. ■ 0 - non-LPDDR5 device in use Present only in designs configured to support LPDDR5 SDRAM memories. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Static
2			**Reserved Field:** Yes
1	lpddr4	R/W	Select LPDDR4 SDRAM ■ 1 - LPDDR4 SDRAM device in use. ■ 0 - non-LPDDR4 device in use Present only in designs configured to support LPDDR4 SDRAM memories. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Static
0			**Reserved Field:** Yes

1.2.2 MSTR2

- **Description:** Master Register2
- **Size:** 32 bits
- **Offset:** 0x10008
- **Exists:** UMCTL2_FREQUENCY_NUM>1

This register is in block REGB_DDRC_CH0.

Table 1-61 Fields for Register: MSTR2

Bits	Name	Memory Access	Description
x:0	target_frequency	R/W	This field specifies the target frequency. ■ 0 - Frequency 0/Normal ■ 1 - Frequency 1/FREQ1 ■ 2 - Frequency 2/FREQ2 ■ 3 - Frequency 3/FREQ3 ■ All other values are reserved. Note: If the target frequency can be changed through Hardware Low Power Interface only, this field is not needed. **Value After Reset:** 0x0 **Exists:** UMCTL2_FREQUENCY_NUM>1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2 **Range Variable[x]:** "DDRCTL_FREQUENCY_BITS" - 1

1.2.3 MSTR4

- **Description:** Master Register4
- **Size:** 32 bits
- **Offset:** 0x10010
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_DDRC_CH0.

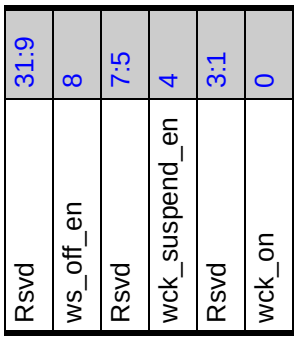


Table 1-62 Fields for Register: MSTR4

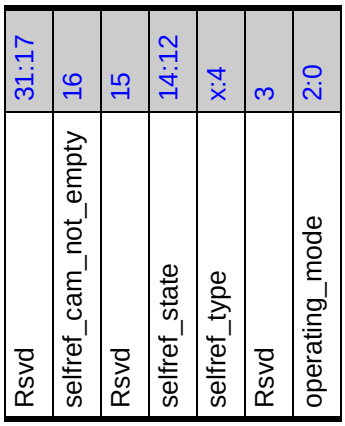
Bits	Name	Memory Access	Description
31:9			**Reserved Field:** Yes
8	ws_off_en	R/W	CAS-WS_OFF enable. If this bit is set to 1, the controller actively issues CAS-WS_OFF command. This register is valid only if MSTR4.wck_on is set to 1. **Value After Reset:** 0x1 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2
7:5			**Reserved Field:** Yes
4	wck_suspend_en	R/W	Enhanced WCK always on mode. If this register is set to 1, the controller issues CAS-WCK_SUSPEND command. This register is valid only if MSTR4.wck_on is set to 1. Note: CAS-WCK_SUSPEND command is valid only when MR0 OP[2]=1b(Enhanced WCK Always On mode supported) and MR18 OP[4]=1b(WCK Always On mode enabled). **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2
3:1			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
0	wck_on	R/W	WCK always ON mode ■ 0: WCK Always On mode disabled ■ 1: WCK Always On mode enabled In case of multi-rank system, the controller issues CAS-WS_FS to all ranks to sets DRAM in sync state simultaneously. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2

1.2.4 STAT

- **Description:** Operating Mode Status Register
- **Size:** 32 bits
- **Offset:** 0x10014
- **Exists:** Always

This register is in block REGB_DDRC_CH0.



31:17	16	15	14:12	x:4	3	2:0
Rsvd	selfref_cam_not_empty	Rsvd	selfref_state	selfref_type	Rsvd	operating_mode

Table 1-63 Fields for Register: STAT

Bits	Name	Memory Access	Description
31:17			**Reserved Field:** Yes
16	selfref_cam_not_empty	R	Self refresh with CAMs not empty. Set to 1 when Self Refresh is entered but CAMs are not drained. Cleared after exiting Self Refresh. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
15			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
14:12	selfref_state	R	Self refresh state. This indicates self refresh or self refresh power down state or Deep Sleep Mode (LPDDR5 only). This register is used for frequency change and MRR/MRW access during self refresh. ■ 000 - SDRAM is not in Self Refresh. ■ 001 - Self refresh 1 ■ 010 - Self refresh power down ■ 011 - Self refresh 2 ■ 100 - Deep Sleep Mode (LPDDR5 only) **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Static
x:4	selfref_type	R	Flags if Self Refresh (except LPDDR4/5) or SR-Powerdown (LPDDR4/5) is entered and if it was under Automatic Self Refresh control only or not. ■ 00 - SDRAM is not in Self Refresh (except LPDDR4/5) or SR-Powerdown (LPDDR4/5). If CA parity retry is enabled by RETRYCTL0.capar_retry_enable, this also indicates SRE command is still in parity error window or retry is in-progress. ■ 11 - SDRAM is in Self Refresh (except LPDDR4/5) or SR-Powerdown (LPDDR4/5), which was caused by Automatic Self Refresh only. If retry is enabled, this guarantees SRE command is executed correctly without parity error. ■ 10 - SDRAM is in Self Refresh (except LPDDR4/5) or SR-Powerdown (LPDDR4/5), which was not caused solely under Automatic Self Refresh control. It could have been caused by Hardware Low Power Interface and/or Software (PWRCTL.selfref_sw). If retry is enabled, this guarantees SRE command is executed correctly without parity error. ■ 01 - SDRAM is in Self Refresh, which was caused by PHY Master Request or Normal PPT2. For LPDDR54 and DDR4, only bit[5:4] are used. For DDR5, self-refresh per rank control is supported. ■ bit[5:4] - rank 0 selfref_type ■ bit[7:6] - rank 1 selfref_type ■ bit[9:8] - rank 2 selfref_type ■ bit[11:10] - rank 3 selfref_type **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static **Range Variable[x]:** "(DDRCTL_DDR_EN==1) ? (MEMC_NUM_RANKS*2) : 2" + 3
3			**Reserved Field:** Yes

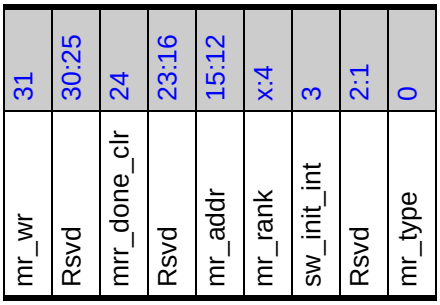
Bits	Name	Memory Access	Description
2:0	operating_mode	R	Operating mode. **DDR4/DDR5 designs:** ■ 000 - Init ■ 001 - Normal ■ 010 - Power-down (For DDR4, this means all ranks are in power-down state. For DDR5, this means at least one rank is in power-down state, check powerdown_state for details) ■ 011 - Self refresh (For DDR4/DDR5, this means all ranks are in self refresh state, check selfref_type for details) ■ 1XX - Maximum Power Saving Mode (For DDR4 only) **LPDDR4/LPDDR5designs:** ■ 000 - Init ■ 001 - Normal ■ 010 - Power-down ■ 011 - Self refresh / Self refresh power-down **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.2.5 MRCTRL0

- **Description:** Mode Register Read/Write Control Register 0.
- **Size:** 32 bits
- **Offset:** 0x10080
- **Exists:** Always

This register is in block REGB_DDRC_CH0.
Note: Do not enable more than one of the following fields simultaneously:

- sw_init_int
- pda_en (DDR4 only field)
- mpr_en (DDR4 only field)
- ppr_en (DDR4 only field)
- ppr_pgmpst_en (DDR4 only field)



31	30:25	24	23:16	15:12	x:4	3	2:1	0
mr_wr	Rsvd	mrr_done_clr	Rsvd	mr_addr	mr_rank	sw_init_int	Rsvd	mr_type

Table 1-64 Fields for Register: MRCTRL0

Bits	Name	Memory Access	Description
31	mr_wr	R/W1S	Setting this register bit to 1 triggers a mode register read or write operation. When the MR operation is complete, the DDRCTL automatically clears this bit. The other register fields of this register must be written in a separate APB transaction, before setting this mr_wr bit. It is recommended NOT to set this signal if in Init, Deep power-down or MPSM operating modes. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
30:25			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
24	mrr_done_clr	R/W1C	If this bit is set, mrr_done will be cleared by the controller. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Testable:** readOnly **Programming Mode:** Dynamic
23:16			**Reserved Field:** Yes
15:12	mr_addr	R/W	Address of the mode register that is to be written to. ■ 0000 - MR0 ■ 0001 - MR1 ■ 0010 - MR2 ■ 0011 - MR3 ■ 0100 - MR4 ■ 0101 - MR5 ■ 0110 - MR6 ■ 0111 - MR7 This signal is also used for writing to control words of the register chip on RDIMMs/LRDIMMs. In that case, it corresponds to the bank address bits sent to the RDIMM/LRDIMM. In case of DDR4, the bit[3:2] corresponds to the bank group bits. Therefore, the bit[3] as well as the bit[2:0] must be set to an appropriate value which is considered both the Address Mirroring of UDIMMs/RDIMMs/LRDIMMs and the Output Inversion of RDIMMs/LRDIMMs. Don't Care for LPDDR4/5 (see MRCTRL1.mr_data for mode register addressing in LPDDR4/5). Don't Care for DDR5 (see CMDCTL.cmd_ctrl for MRW/MRR access in DDR5). **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
x:4	mr_rank	R/W	Controls which rank is accessed by MRCTRL0.mr_wr. Normally, it is desired to access all ranks, so all bits must be set to 1. However, for multi-rank UDIMMs/RDIMMs/LRDIMMs which implement address mirroring, it may be necessary to access ranks individually. Examples (assume DDRCTL is configured for 4 ranks): ■ 0x1 - select rank 0 only ■ 0x2 - select rank 1 only ■ 0x5 - select ranks 0 and 2 ■ 0xA - select ranks 1 and 3 ■ 0xF - select ranks 0, 1, 2 and 3 Don't Care for DDR5. **Value After Reset:** "(MEMC_NUM_RANKS==4) ? 0xF :((MEMC_NUM_RANKS==2) ? 0x3 : 0x1)" **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_NUM_RANKS" + 3
3	sw_init_int	R/W	Indicates whether Software intervention is allowed via MRCTRL0/MRCTRL1 before automatic SDRAM initialization routine or not. For DDR4, this bit can be used to initialize the DDR4 RCD (MR7) before automatic SDRAM initialization. For LPDDR4/5, this bit can be used to program additional mode registers before automatic SDRAM initialization if necessary. In LPDDR4 dual channel mode, note that this must be programmed to both channels beforehand. Note that this must be cleared to 0 after completing Software operation. Otherwise, SDRAM initialization routine will not re-start. ■ 0 - Software intervention is not allowed ■ 1 - Software intervention is allowed Don't Care for DDR5. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
2:1			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
0	mr_type	R/W	Indicates whether the mode register operation is read or write. ■ 0 - Write ■ 1 - Read Only used for LPDDR4/LPDDR5/DDR4. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.2.6 MRCTRL1

- **Description:** Mode Register Read/Write Control Register 1
- **Size:** 32 bits
- **Offset:** 0x10084
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

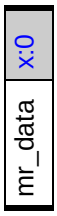


Table 1-65 Fields for Register: MRCTRL1

Bits	Name	Memory Access	Description
x:0	mr_data	R/W	Mode register write data for DDR4 mode. For LPDDR4/5, MRCTRL1[15:0] are interpreted as ■ [15:8] MR Address ■ [7:0] MR data for writes, don't care for read This is 18-bit wide in configurations with DDR4 support and 16-bits for the LPDDR5/4 controller. For DDR4 PPR, this is used for row address field in the ACT command of the PPR sequence. Don't Care for DDR5 (see CMDCTL.cmd_ctrl for MRW access in DDR5). **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_PAGE_BITS" - 1

1.2.7 MRSTAT

- **Description:** Mode Register Read/Write Status Register
- **Size:** 32 bits
- **Offset:** 0x10090
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

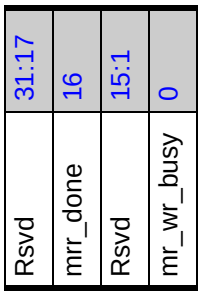


Table 1-66 Fields for Register: MRSTAT

Bits	Name	Memory Access	Description
31:17			**Reserved Field:** Yes
16	mrr_done	R	This signal goes high when the controller received MRR data which is triggered by MRCTRL0.mr_wr. This signal is cleared by mrr_done_clr **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic
15:1			**Reserved Field:** Yes
0	mr_wr_busy	R	The SoC core may initiate a MR write operation only if this signal is low. This signal goes high in the clock after the DDRCTL accepts the MRW/MRR request. It goes low when the MRW/MRR command is issued to the SDRAM. It is recommended not to perform MRW/MRR commands when 'MRSTAT.mr_wr_busy' is high. ■ 0 - Indicates that the SoC core can initiate a mode register write operation ■ 1 - Indicates that mode register write operation is in progress **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.2.8 MRRDATA0

- **Description:** Mode Register Read Data 0
- **Size:** 32 bits
- **Offset:** 0x10094
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_DDRC_CH0.

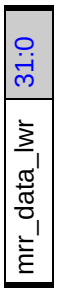


Table 1-67 Fields for Register: MRRDATA0

Bits	Name	Memory Access	Description
31:0	mrr_data_lwr	R	MRR data for DQ[31:0] This register is updated when the controller issued MRR command triggered by MRCTRL register. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic

1.2.9 MRRDATA1

- **Description:** Mode Register Read Data 1
- **Size:** 32 bits
- **Offset:** 0x10098
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_DDRC_CH0.

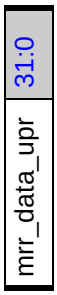


Table 1-68 Fields for Register: MRRDATA1

Bits	Name	Memory Access	Description
31:0	mrr_data_upr	R	MRR data for DQ[63:32] This register is updated when the controller issued MRR command triggered by MRCTRL register. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic

1.2.10 DERATECTL0

- **Description:** Temperature Derate Control Register 0
- **Size:** 32 bits
- **Offset:** 0x10100
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_DDRC_CH0.

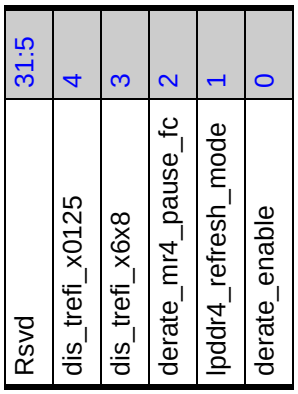


Table 1-69 Fields for Register: DERATECTL0

Bits	Name	Memory Access	Description
31:5			**Reserved Field:** Yes
4	dis_trefi_x0125	R/W	Disables 0.125 x tREFI refresh rate for derating. When this register filed is set to 1, controller behaves like 0.25 x tREFI refresh rate mode although in 0.125 x tREFI refresh rate mode. And controller asserts interrupt signal "derate_temp_limit_intr" when receives MR4 OP[4:0] = 01110 or 01111. ■ 0 - Enable 0.125 x tREFI refresh rate ■ 1 - Disable 0.125 x tREFI refresh rate Note: This register field is only applicable for designs supporting LPDDR5 SDRAM. This register bit is required to set 1 if per-bank refresh "RFSHMOD0.per_bank_refresh=1" and derating "DERATECTL0.derate_enable=1" are enabled. Contact synopsys for more information. **Value After Reset:** 0x1 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
3	dis_trefi_x6x8	R/W	Disables 8x tREFI and 6x tREFI refresh rate for derating.When this register filed is set to 1, controller behaves like 4x tREFI refresh rate mode even though in 8x tREFI and 6x tREFI mode. ■ 0 - Enable 6x tREFI and 8x tREFI refresh rate ■ 1 - Disable 6x tREFI and 8x tREFI refresh rate Note: This register field is only applicable for designs supporting LPDDR5 SDRAM. This register bit is required to set 1 if LPDDR5 is used. Contact synopsys for more information. **Value After Reset:** 0x1 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic - Refresh Related
2	derate_mr4_pause_fc	R/W	Pauses automatic MRR to MR4. For more details, see description of DERATECTL0.derate_enable. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
1	lpddr4_refresh_mode	R/W	Selects the LPDDR4 refresh mode ■ 0 - Legacy refresh mode ■ 1 - Modified refresh mode (Unsupported) **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Static
0	derate_enable	R/W	Enables derating ■ 0 - Timing parameter derating is disabled ■ 1 - Timing parameter derating is enabled using MR4 read value. Note that, once DERATECTL0.derate_enable is set to 1, it has to keep 1. Otherwise, the refresh rate and other timing parameters revert to their nominal values. To stop automatic MRR to MR4 temporarily after setting DERATECTL0.derate_enable =1, DERATECTL0.derate_mr4_pause_fc needs to be set to 1 without changing DERATECTL0.derate_enable. Setting DERATECTL0.derate_mr4_pause_fc=0 without changing DERATECTL0.derate_enable restarts automatic MRR to MR4. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic

1.2.11 DERATECTL1

- **Description:** Temperature Derate Control Register 1
- **Size:** 32 bits
- **Offset:** 0x10104
- **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1

This register is in block REGB_DDRC_CH0.

Table 1-70 Fields for Register: DERATECTL1

Bits	Name	Memory Access	Description
x:0	active_derate_byte_rank0	R/W	Indicates which byte of the MRR data is used for derating in rank0. The each bit corresponds each byte. If the multiple register bits are enabled, controller compares refresh rate of the corresponding devices and chooses the worst refresh rate among them This register only supports LPDDR4, LPDDR5 and DDR5. For LPDDR4 and LPDDR5: Valid width is MEMC_DRAM_DATA_WIDTH/8. This bit[n]=1 means that DQ[8*n+:8] is valid MRR data. All "0"s is invalid, if DERATECTL0.derate_enable=1. For DDR5: Valid width is MEMC_DRAM_TOTAL_DATA_WIDTH/device DQ width. Device DQ width is based on MSTR0.device_config register value. Note: When data_bus_width Half or Quarter, only half or Quarter of MEMC_DRAM_TOTAL_DATA_WIDTH are valid. Note: In Dual Channel configuration, this register apply to both channels, and each channel take in charge its DQ respectively. Note: Derating is not applicable for DDR5/4 and any reference to DERATECTL0 can be ignored. DDR5 Refresh rate is changed according to MR4 that is periodically read by the controller. The values of those registers must be set assuming the MR4 register read from the controller. **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Programming Mode:** Static **RangeVariable[x]:**"MEMC_DRAM_TOTAL_DATA_WIDTH/4" - 1

1.2.12 DERATECTL2

- **Description:** Temperature Derate Control Register 2
- **Size:** 32 bits
- **Offset:** 0x10108
- **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 && MEMC_NUM_RANKS>1

This register is in block REGB_DDRC_CH0.

Table 1-71 Fields for Register: DERATECTL2

Bits	Name	Memory Access	Description
x:0	active_derate_byte_rank1	R/W	Indicates which byte of the MRR data is used for derating in rank1. The each bit corresponds each byte. If the multiple register bits are enabled, controller compares refresh rate of the corresponding devices and chooses the worst refresh rate among them. This register only supports LPDDR4, LPDDR5 and DDR5. For LPDDR4 and LPDDR5 : Valid width is MEMC_DRAM_DATA_WIDTH/8. This bit[n]=1 means that DQ[8*n+:8] is valid MRR data. All "0"s is invalid, if DERATECTL0.derate_enable=1. For DDR5 : Valid width is MEMC_DRAM_TOTAL_DATA_WIDTH/device DQ width. Device DQ width is based on MSTR0.device_config register value. Note: When data_bus_width Half or Quarter, only half or Quarter of MEMC_DRAM_TOTAL_DATA_WIDTH are valid. Note: In Dual Channel configuration, this register apply to both channels, and each channel take in charge its DQ respectively. **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 && MEMC_NUM_RANKS>1 **Programming Mode:** Static **RangeVariable[x]:**"MEMC_DRAM_TOTAL_DATA_WIDTH/4" - 1

1.2.13 DERATECTL5

- **Description:** Temperature Derate Control Register 5
- **Size:** 32 bits
- **Offset:** 0x10114
- **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1

This register is in block REGB_DDRC_CH0.

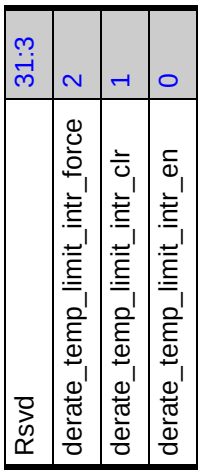


Table 1-72 Fields for Register: DERATECTL5

Bits	Name	Memory Access	Description
31:3			**Reserved Field:** Yes
2	derate_temp_limit_intr_force	R/W1C	Interrupt force bit for derate_temp_limit_intr. Setting this register to 1 will cause the derate_temp_limit_intr output pin to be asserted. At the end of the interrupt force operation, the DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Testable:** readOnly **Programming Mode:** Dynamic
1	derate_temp_limit_intr_clr	R/W1C	Interrupt clear bit for derate_temp_limit_intr. At the end of the interrupt clear operation, the DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
0	derate_temp_limit_intr_en	R/W	Interrupt enable bit for derate_temp_limit_intr output pin. ■ 1 Enabled ■ 0 Disabled **Value After Reset:** 0x1 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Programming Mode:** Dynamic

1.2.14 DERATECTL6

- **Description:** Temperature Derate Control Register 6
- **Size:** 32 bits
- **Offset:** 0x10118
- **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1

This register is in block REGB_DDRC_CH0.

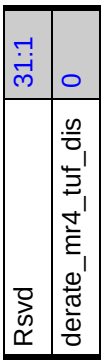


Table 1-73 Fields for Register: DERATECTL6

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	derate_mr4_tuf_dis	R/W	Disable use of MR4 TUF flag (MR4[7]) bit. ■ 0 - Use MR4 TUF flag (MR4[7]) ■ 1 - Do not use MR4 TUF Flag (MR4[7]) It is recommended to set this register to 1. This affects both the periodic refresh rate update and asserting interrupt signal derate_temp_limit_intr. (i.e. In derate_mr4_tuf_dis==1, the controller can update the refresh rate, and assert the derate_temp_limit_intr if it exceeds the thresholds irrespective of the value of TUF flag.) **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

1.2.15 DERATESTAT0

- **Description:** Temperature Derate Status Register 0
- **Size:** 32 bits
- **Offset:** 0x1011c
- **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1

This register is in block REGB_DDRC_CH0.

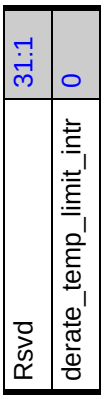


Table 1-74 Fields for Register: DERATESTAT0

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	derate_temp_limit_intr	R	Derate temperature interrupt indicating SDRAM temperature operating limit is exceeded. In LPDDR4, this register field is set to 1 when the value read from MR4[2:0] is 3'b000 or 3'b111. In LPDDR5, this register field is set to 1 when the value read from MR4[4:0] is 5'b00000 or 5'b11111 or invalid value. In DDR5, this register field is set to 1 when the value read from MR4[2:0] is the thresholds programmed by DERATECTL2.derate_low_temp_limit and DERATECTL2.derate_high_temp_limit. Cleared by register DERATECTL1.derate_temp_limit_intr_clr. **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Testable:** readOnly **Programming Mode:** Static

1.2.16 DERATEDBGCTL

- **Description:** Temperature Derate Debug Contrl Register
- **Size:** 32 bits
- **Offset:** 0x10124
- **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1

This register is in block REGB_DDRC_CH0.

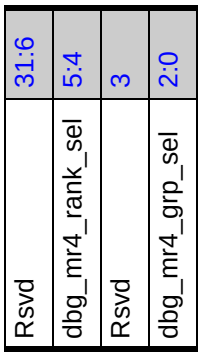


Table 1-75 Fields for Register: DERATEDBGCTL

Bits	Name	Memory Access	Description
31:6			**Reserved Field:** Yes
5:4	dbg_mr4_rank_sel	R/W	MR4 rank select in case of multi ranks **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Programming Mode:** Static
3			**Reserved Field:** Yes
2:0	dbg_mr4_grp_sel	R/W	MR4 data group select based on 4 device MRR read data **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Programming Mode:** Static

1.2.17 DERATEDBGSTAT

- **Description:** Temperature Derate Debug Status Register
- **Size:** 32 bits
- **Offset:** 0x10128
- **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1

This register is in block REGB_DDRC_CH0.

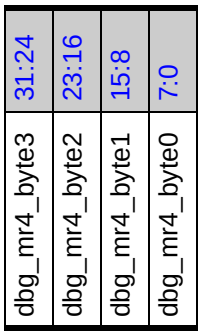


Table 1-76 Fields for Register: DERATEDBGSTAT

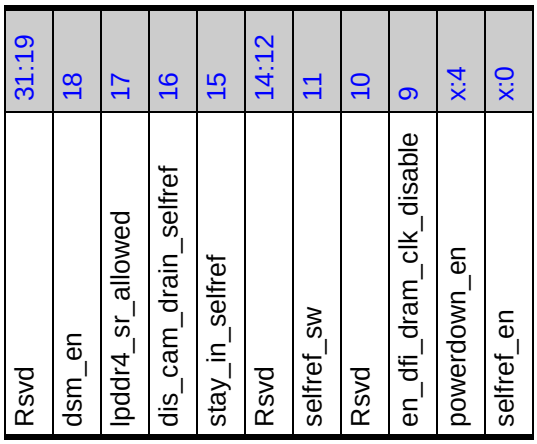
Bits	Name	Memory Access	Description
31:24	dbg_mr4_byte3	R	Byte 3 of selected 32 bits MR4 data. LPDDR4/LPDDR5 use 5 bits and DDR5 uses 8 bits. This register is updated when the controller issues MRR for MR4 automatically. For LPDDR4 and LPDDR5 : It is occurred only if derating is enabled. (DERATECTL0.derate_enable=1) **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Programming Mode:** Dynamic
23:16	dbg_mr4_byte2	R	Byte 2 of selected 32 bits MR4 data. LPDDR4/LPDDR5 use 5 bits and DDR5 uses 8 bits. This register is updated when the controller issues MRR for MR4 automatically. For LPDDR4 and LPDDR5 : It is occurred only if derating is enabled. (DERATECTL0.derate_enable=1) **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Programming Mode:** Dynamic
15:8	dbg_mr4_byte1	R	Byte 1 of selected 32 bits MR4 data. LPDDR4/LPDDR5 use 5 bits and DDR5 uses 8 bits. This register is updated when the controller issues MRR for MR4 automatically. For LPDDR4 and LPDDR5 : It is occurred only if derating is enabled. (DERATECTL0.derate_enable=1) **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
7:0	dbg_mr4_byte0	R	Byte 0 of selected 32 bits MR4 data. LPDDR4/LPDDR5 use 5 bits and DDR5 uses 8 bits. This register is updated when the controller issues MRR for MR4 automatically. For LPDDR4 and LPDDR5 : It is occurred only if derating is enabled. (DERATECTL0.derate_enable=1) **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Programming Mode:** Dynamic

1.2.18 PWRCTL

- **Description:** Low Power Control Register
- **Size:** 32 bits
- **Offset:** 0x10180
- **Exists:** Always

This register is in block REGB_DDRC_CH0.



31:19	18	17	16	15	14:12	11	10	9	x:4	x:0
Rsvd	dsm_en	lpddr4_sr_allowed	dis_cam_drain_selfref	stay_in_selfref	Rsvd	selfref_sw	Rsvd	en_dfi_dram_clk_disable	powerdown_en	selfref_en

Table 1-77 Fields for Register: PWRCTL

Bits	Name	Memory Access	Description
31:19			**Reserved Field:** Yes
18	dsm_en	R/W	A value of 1 to this register causes system to move to Deep Sleep Mode state immediately. ■ 1 - Entry to Deep Sleep Mode ■ 0 - Exit from Deep Sleep Mode **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic
17	lpddr4_sr_allowed	R/W	Indicates whether transition from SR-PD to SR and back to SR-PD is allowed. This register should be set to '1' if any of PHYMSTR or PPT features is enabled. This register field cannot be modified while PWRCTL.selfref_sw==1. ■ 0 - SR-PD -> SR -> SR-PD not allowed ■ 1 - SR-PD -> SR -> SR-PD allowed **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
16	dis_cam_drain_selfref	R/W	Indicates whether skipping CAM draining is allowed when entering Self-Refresh. This register field cannot be modified while PWRCTL.selfref_sw==1. ■ 0 - CAMs must be empty before entering SR ■ 1 - CAMs are not emptied before entering SR (unsupported) Note, PWRCTL.dis_cam_drain_selfref=1 is unsupported in this release. PWRCTL.dis_cam_drain_selfref=0 is required. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
15	stay_in_selfref	R/W	Self refresh state is an intermediate state to enter to Self refresh power down state or exit Self refresh power down state for LPDDR4/5. This register controls transition from the Self refresh state. ■ 1 - Prohibit transition from Self refresh state ■ 0 - Allow transition from Self refresh state **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic
14:12			**Reserved Field:** Yes
11	selfref_sw	R/W	A value of 1 to this register causes system to move to Self Refresh state immediately, as long as it is not in INIT or DPD/MPSM operating mode. This is referred to as Software Entry/Exit to Self Refresh. ■ 1 - Software Entry to Self Refresh ■ 0 - Software Exit from Self Refresh **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
10			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
9	en_dfi_dram_clk_disable	R/W	Enable the assertion of dfi_dram_clk_disable whenever a clock is not required by the SDRAM. If set to 0, dfi_dram_clk_disable is never asserted. Assertion of dfi_dram_clk_disable is as follows: In DDR4, can be asserted in following: ■ in Self Refresh ■ in Maximum Power Saving Mode In LPDDR4/LPDDR5, can be asserted in following: ■ in Self Refresh Power Down ■ in Power Down ■ during Normal operation (Clock Stop) In DDR5, can be asserted in following: ■ in Self Refresh In DDR5 (L)RDIMM, the value of this field need to be same as DIMMCTL.dimm_selfref_clock_stop_mode. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
x:4	powerdown_en	R/W	If true then the DDRCTL goes into power-down after a programmable number of cycles "maximum idle clocks before power down" (PWRTMG.powerdown_to_x32). This register bit may be re-programmed during the course of normal operation. For LPDDR4/5 and DDR4, only bit[4] is used. For DDR5, powerdown per rank enable is supported. ■ bit[4] - rank 0 powerdown_en ■ bit[5] - rank 1 powerdown_en ■ bit[6] - rank 2 powerdown_en ■ bit[7] - rank 3 powerdown_en **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "(DDRCTL_DDR_EN==1) ? MEMC_NUM_RANKS : 1" + 3

Bits	Name	Memory Access	Description
x:0	selfref_en	R/W	If true then the DDRCTL puts the SDRAM per rank into Self Refresh after a programmable number of cycles "maximum idle clocks before Self Refresh (PWRTMG.selfref_to_x32)". This register bit may be re-programmed during the course of normal operation. For LPDDR4/5 and DDR4, only bit[0] is used. For DDR5, self-refresh per rank enable is provided. Current self-refresh need to be enabled for all ranks. For DDR5 (L)RDIMM, self-refresh need to be enabled for all ranks of both channels. ■ bit[0] - rank 0 selfref_en ■ bit[1] - rank 1 selfref_en ■ bit[2] - rank 2 selfref_en ■ bit[3] - rank 3 selfref_en **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "(DDRCTL_DDR_EN==1) ? MEMC_NUM_RANKS : 1" - 1

1.2.19 HWLPCTL

- **Description:** Hardware Low Power Control Register
- **Size:** 32 bits
- **Offset:** 0x10184
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

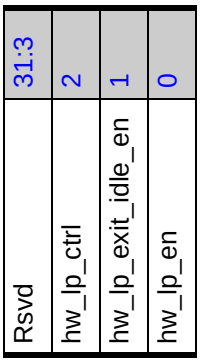


Table 1-78 Fields for Register: HWLPCTL

Bits	Name	Memory Access	Description
31:3			**Reserved Field:** Yes
2	hw_lp_ctrl	R/W	■ 1 - Channel 0 hardware low power interface controls self refresh requests to both channel 0 and 1 devices. Channel 1 hardware low power interface is not used in this case. HWLPCTL.hw_lp_exit_idle_en & HWLPCTL.hw_lp_en from both channels should be set to identical values in this case. ■ 0 - Channel 0/1 hardware low power interface control self refresh requests to channel 0/1 devices respectively. This is only present for dual channel configurations. hw_lp_ctrl should be set to 1 for DDR5 (L)RDIMM. **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_DUAL_CHANNEL__OR__SINGLE_INST_DUALCH==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
1	hw_lp_exit_idle_en	R/W	When this bit is programmed to 1 the cactive_in_ddrc pin of the DDRC can be used to exit from the automatic clock stop, automatic power down or automatic self-refresh modes. Note, it will not cause exit of Self-Refresh that was caused by Hardware Low Power Interface and/or Software (PWRCTL.selfref_sw). **Value After Reset:** 0x1 **Exists:** Always **Testable:** readOnly **Programming Mode:** Static
0	hw_lp_en	R/W	Enable for Hardware Low Power Interface. **Value After Reset:** 0x1 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2

1.2.20 CLKGATECTL

- **Description:** clock gate control
- **Size:** 32 bits
- **Offset:** 0x1018c
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_DDRC_CH0.

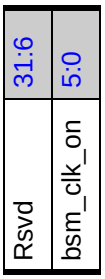


Table 1-79 Fields for Register: CLKGATECTL

Bits	Name	Memory Access	Description
31:6			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
5:0	bsm_clk_on	R/W	Indicates whether the output signal bsm_clk_en [MEMC_NUM_RANKS-1:0] become 0 for each corresponding DDRCTL internal state: ■ 0: bsm_clk_en become 0 in case where DDRCTL is in corresponding state described below ■ 1: bsm_clk_en remain 1 in case where DDRCTL is in corresponding state The bsm_clk_en [MEMC_NUM_RANKS-1:0] indicates that clock can be removed when corresponding rank of this signal is 0. Each corresponding DDRCTL internal state is as follows: ■ [0] Unpopulated rank control ■ [1] Controller initialization state (until dfi0_init_start/dfi0_init_complete handshake is done) ■ [2] Self Refresh mode ■ [3] Self Refresh Powerdown mode ■ [4] Powerdown mode ■ [5] Deep Sleep Mode (LPDDR5 Only) bit[5:1] indicates behavior of the bsm_clk_en for each corresponding DDRCTL internal state. For example, if this field is set to 6'b00_0100, bsm_clk_en becomes 1 when it is in self refresh mode. This implies that bsm_clk is not removed while it is in self refresh mode, but the bsm_clk is removed by external clock gating logic in other modes above. If bit 0 (Unpopulated rank control) is set to 1, behavior of the bsm_clk_en is determined by other fields in this register irrespective of MSTR.active_ranks. If the bsm_clk_en[1]=0 is needed in case of single rank (MSTR.active_ranks=1), bit 0 has to be set to 0. To maximize power-saving, this field needs to be set to 6'b00_0000. **Value After Reset:** 0x3f **Exists:** Always **Programming Mode:** Static

1.2.21 RFSHMOD0

- **Description:** Refresh Mode Register 0
- **Size:** 32 bits
- **Offset:** 0x10200
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

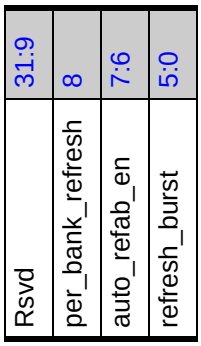


Table 1-80 Fields for Register: RFSHMOD0

Bits	Name	Memory Access	Description
31:9			**Reserved Field:** Yes
8	per_bank_refresh	R/W	■ 1 - Per bank refresh ■ 0 - All bank refresh Per bank refresh allows traffic to flow to other banks. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
7:6	auto_refab_en	R/W	Enables automatic switching from per-bank to all-bank refresh when the derated refresh period is small. ■ 0 - Disable automatic switching ■ 1 - Enable automatic switching when the derated refresh period is tREFI/2 or lower ■ 2 - Enable automatic switching when the derated refresh period is tREFI/4 or lower ■ 3 - Enable automatic switching when the derated refresh period is tREFI/8 Automatically switches back to per-bank refresh when the derated refresh period is no longer small. This register should be programmed to 2'b00 if RFSHMOD0.per_bank_refresh = 1'b0. This register should be programmed to 2'b00 for LPDDR4 2Gb and 4Gb. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
5:0	refresh_burst	R/W	The programmed value + 1 is the number of refresh timeouts that is allowed to accumulate before traffic is blocked and the refreshes are forced to execute. Closing pages to perform a refresh is a one-time penalty that must be paid for each group of refreshes. Therefore, performing refreshes in a burst reduces the per-refresh penalty of these page closings. Higher numbers for RFSHCTL.refresh_burst slightly increases utilization; lower numbers decreases the worst-case latency associated with refreshes. ■ 0 - single refresh ■ 1 - burst-of-2 refresh ■ 7 - burst-of-8 refresh In DDR4 mode, according to Fine Granularity feature, 8 refreshes can be postponed in 1X mode, 16 refreshes in 2X mode and 32 refreshes in 4X mode. In DDR5 mode, according to Fine Granularity feature, 4 refreshes can be postponed in 1X mode and 8 refreshes can be postponed in 2X mode. In DDR5 mode, if self-refresh operation is expected, then this field shall not be set to the maximum number, for example, according to Fine Granularity feature, it should be smaller than 8 in 2X mode. In per-bank refresh mode of LPDDR4/5 (RFSHMOD0.per_bank_refresh = 1), 56 refreshes can be postponed. If using PHY-initiated updates or PPT2 (LPDDR only), care must be taken in the setting of RFSHMOD0.refresh_burst, to ensure that tRFCmax and tREFI are not violated due to PHY-initiated updates or PPT2 occurring shortly before a refresh burst was due. In this situation, the refresh burst will be delayed until they complete. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic - Refresh Related

1.2.22 RFSHCTL0

- **Description:** Refresh Control Register 0
- **Size:** 32 bits
- **Offset:** 0x10208
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

Table 1-81 Fields for Register: RFSHCTL0

Bits	Name	Memory Access	Description
31:5			**Reserved Field:** Yes
4	refresh_update_level	R/W	Toggle this signal (either from 0 to 1 or from 1 to 0) to indicate that the refresh register(s) have been updated. refresh_update_level must not be toggled when the DDRC is in reset (core_ddrc_rstn = 0). In DDR5 mode, this can be toggled during self-refresh mode and MPSM in OPS state. The refresh register(s) are automatically updated when exiting reset. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
3:1			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
0	dis_auto_refresh	R/W	When '1', disable auto-refresh generated by the DDRCTL. When auto-refresh is disabled, the SoC core must generate refreshes using the registers OPREFCTRL*.rankn_refresh. When dis_auto_refresh transitions from 0 to 1, any pending refreshes are immediately scheduled by the DDRCTL. If DDR4 CA parity retry is enabled (RETRYCTL0.capar_retry_enable = 1), disable auto-refresh is not supported, and this bit must be set to '0'. If FGR mode is enabled (RFSHMOD1.fgr_mode > 0), disable auto-refresh is not supported, and this bit must be set to '0'. This register field is changeable on the fly in non-DDR5 mode, and changeable during INIT/DBG/BIST state or self-refresh mode and MPSM during OPS state in DDR5 mode. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic - Refresh Related

1.2.23 RFMMOD0

- **Description:** RFM Mode Register 0
- **Size:** 32 bits
- **Offset:** 0x10220
- **Exists:** DDRCTL_LPDDR_RFM==1

This register is in block REGB_DDRC_CH0.

31:29	28:24	23:20	19:18	17:16	15:13	12:8	7:5	4	3:1	0
Rsvd	rfmth_rm_thr	Rsvd	raadec	raamult	Rsvd	raaimt	Rsvd	rfmsbc	Rsvd	rfm_en

Table 1-82 Fields for Register: RFMMOD0

Bits	Name	Memory Access	Description
31:29			**Reserved Field:** Yes
28:24	rfmth_rm_thr	R/W	Threshold of RM (Refresh Multiplier) to disable RFM command. When current RM which is read from MR4:OP[4:0] in LPDDR5 SDRAM is greater than or equal to this, RFM command is disabled irrespective of RAA count. This is calculated from tREFIe <= RFMTH requirement. **Value After Reset:** 0xa **Exists:** DDRCTL_LPDDR_RFM==1 **Programming Mode:** Static
23:20			**Reserved Field:** Yes
19:18	raadec	R/W	RAADEC: RAA Count Decrement per RFM Command as programmed in MR57:OP[1:0] of LPDDR5. ■ 2'b00: RAAIMT ■ 2'b01: RAAIMT * 1.5 ■ 2'b10: RAAIMT * 2 ■ 2'b11: RAAIMT * 4 **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR_RFM==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
17:16	raamult	R/W	RAAMULT: Rolling Accumulated ACT Multiplier as programmed in MR27:OP[7:6] of LPDDR5. ■ 2'b00: 2X ■ 2'b01: 4X ■ 2'b10: 6X ■ 2'b11: 8X **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR_RFM==1 **Programming Mode:** Static
15:13			**Reserved Field:** Yes
12:8	raaimt	R/W	RAAIMT: Rolling Accumulated ACT Initial Management Threshold as programmed in MR27:OP[5:1] of LPDDR5. ■ 5'b00000: Invalid ■ 5'b00001: 8 ■ 5'b00010: 16 : ■ 5'b11110: 240 ■ 5'b11111: 248 **Value After Reset:** 0x1 **Exists:** DDRCTL_LPDDR_RFM==1 **Programming Mode:** Static
7:5			**Reserved Field:** Yes
4	rfmsbc	R/W	RFMSBC: Single-Bank Counters Implemented. The value should be determined based on MR57:OP[5:4] of LPDDR5. ■ 1'b0: One RAA counter per two banks (1 of 8), MR57:OP[5:4]==2'b00 ■ 1'b1: One RAA counter per one bank (1 of 16), MR57:OP[5:4]==2'b01 **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR_RFM==1 && DDRCTL_LPDDR_RFMSBC==1 **Programming Mode:** Static
3:1			**Reserved Field:** Yes
0	rfm_en	R/W	RFM enable in LPDDR5 mode. This should be programmed based on MR27:OP[0] (RFM Required) of LPDDR5. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR_RFM==1 **Programming Mode:** Static

1.2.24 ZQCTL0

- **Description:** ZQ Control Register 0
- **Size:** 32 bits
- **Offset:** 0x10280
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

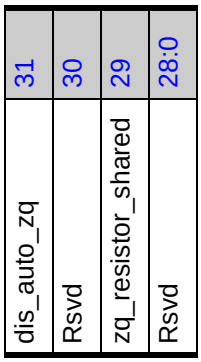


Table 1-83 Fields for Register: ZQCTL0

Bits	Name	Memory Access	Description
31	dis_auto_zq	R/W	■ 1 - Disable DDRCTL generation of ZQCS/MPC(ZQ calibration) command. Register OPCTRLCMD.zq_calib_short can be used instead to issue ZQ calibration request from APB module. ■ 0 - Internally generate ZQCS/MPC(ZQ calibration) commands based on ZQSET1TMG1.t_zq_short_interval_x1024. This register field only applies to DDR4, LPDDR4, and LPDDR5. For DDR5, see PASCTL7~PASCTL10 registers **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
30			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
29	zq_resistor_shared	R/W	■ 1 - Denotes that ZQ resistor is shared between ranks. Means ZQinit/ZQCL/ZQCS/MPC(ZQ calibration) commands are sent to one rank at a time with tZQinit/tZQCL/tZQCS/tZQCAL/tZQLAT timing met between commands so that commands to different ranks do not overlap. ■ 0 - ZQ resistor is not shared. If LPDDR5 is used, this register needs to be set to "0". **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
28:0			**Reserved Field:** Yes

1.2.25 ZQCTL1

- **Description:** ZQ Control Register 1
- **Size:** 32 bits
- **Offset:** 0x10284
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_DDRC_CH0.

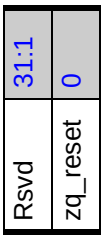


Table 1-84 Fields for Register: ZQCTL1

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	zq_reset	R/W1S	Setting this register bit to 1 triggers a ZQ Reset operation. When the ZQ Reset operation is complete, the DDRCTL automatically clears this bit. It is recommended NOT to set this register bit if in Init, in SR-Powerdown or Deep Sleep Modes. For SR-Powerdown it will be scheduled after SRPD has been exited. For Deep Sleep Mode, it will be scheduled after DSM and/or SRPD has been exited. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Testable:** readOnly **Programming Mode:** Dynamic

1.2.26 ZQCTL2

- **Description:** ZQ Control Register 2
- **Size:** 32 bits
- **Offset:** 0x10288
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

Table 1-85 Fields for Register: ZQCTL2

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	dis_srx_zqcl	R/W	■ 1 - Disable issuing of ZQCL/MPC(ZQ calibration) command at Self-Refresh/SR-Powerdown exit. ■ 0 - Enable issuing of ZQCL/MPC(ZQ calibration) command at Self-Refresh/SR-Powerdown exit. This is only present for designs supporting DDR4 or DDR5 or LPDDR4 or LPDDR5 devices. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2, Group 4

1.2.27 ZQSTAT

- **Description:** ZQ Status Register
- **Size:** 32 bits
- **Offset:** 0x1028c
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_DDRC_CH0.

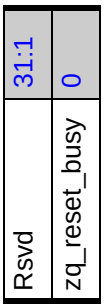


Table 1-86 Fields for Register: ZQSTAT

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	zq_reset_busy	R	SoC core may initiate a ZQ Reset operation only if this signal is low. This signal goes high in the clock after the DDRCTL accepts the ZQ Reset request. It goes low when the ZQ Reset command is issued to the SDRAM and the associated NOP period is over. It is recommended not to perform ZQ Reset commands when this signal is high. ■ 0 - Indicates that the SoC core can initiate a ZQ Reset operation ■ 1 - Indicates that ZQ Reset operation is in progress **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.2.28 DQSOSCRUNTIME

- **Description:** DQS/WCK Oscillator Runtime Register
- **Size:** 32 bits
- **Offset:** 0x10300
- **Exists:** LPDDR45_DQSOSC_EN==1

This register is in block REGB_DDRC_CH0.

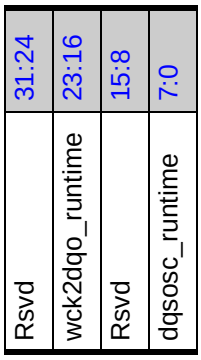


Table 1-87 Fields for Register: DQSOSCRUNTIME

Bits	Name	Memory Access	Description
31:24			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
23:16	wck2dqo_runtime	R/W	WCK2DQO interval timer run time setting as programmed in MR40 for LPDDR5.This field must be non zero. ■ 0x0 - Interval timer stop via MPC command (not supported) ■ 0x1 - Interval timer stops automatically at 16th clocks after timer start ■ 0x2 - Interval timer stops automatically at 32nd clocks after timer start ■ 0x3 - Interval timer stops automatically at 48th clocks after timer start ■ 0x4 - Interval timer stops automatically at 64th clocks after timer start ■ ----- Thru ---------- ■ 0x3F - Interval timer stops automatically at (63x16)th clocks after timer start ■ 0x40 to 0x7F - Interval timer stops automatically at 2048th clocks after timer start ■ 0x80 to 0xBF - Interval timer stops automatically at 4096th clocks after timer start ■ 0xC0 to 0xFF - Interval timer stops automatically at 8192nd clocks after timer start This register field is only applicable for designs supporting LPDDR5 SDRAM memories. It is don't care for LPDDR4 SDRAM memories. Unit: DRAM clock cycles. **Value After Reset:** 0x40 **Exists:** LPDDR45_DQSOSC_EN==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2
15:8			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
7:0	dqsosc_runtime	R/W	LPDDR4: DQS interval timer run time setting as programmed in MR23 LPDDR5: WCK2DQI interval timer run time setting as programmed in MR37 DDR5: DQS interval timer run time setting as programmed in MR45 This field must be non zero. ■ 0x0 - Interval timer stop via MPC command (not supported) ■ 0x1 - Interval timer stops automatically at 16th clocks after timer start ■ 0x2 - Interval timer stops automatically at 32nd clocks after timer start ■ 0x3 - Interval timer stops automatically at 48th clocks after timer start ■ 0x4 - Interval timer stops automatically at 64th clocks after timer start ■ ----- Thru ---------- ■ 0x3F - Interval timer stops automatically at (63x16)th clocks after timer start ■ 0x40 to 0x7F - Interval timer stops automatically at 2048th clocks after timer start ■ 0x80 to 0xBF - Interval timer stops automatically at 4096th clocks after timer start ■ 0xC0 to 0xFF - Interval timer stops automatically at 8192nd clocks after timer start Unit: DRAM clock cycles. **Value After Reset:** 0x40 **Exists:** LPDDR45_DQSOSC_EN==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2

1.2.29 DQSOSCSTAT0

- **Description:** DQS/WCK Oscillator Status Register 0
- **Size:** 32 bits
- **Offset:** 0x10304
- **Exists:** LPDDR45_DQSOSC_EN==1

This register is in block REGB_DDRC_CH0.

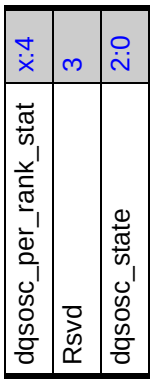


Table 1-88 Fields for Register: DQSOSCSTAT0

Bits	Name	Memory Access	Description
x:4	dqsosc_per_rank_stat	R	DQS/WCK Oscillator per rank status. This bit is set to 0 when DQSOSCCTL0.dqsosc_enable is set to 1, and set to 1 when the DQS Oscillator command sequence is started for the corresponding active rank. **Value After Reset:** 0x0 **Exists:** LPDDR45_DQSOSC_EN==1 **Programming Mode:** Static **Range Variable[x]:** MEMC_NUM_RANKS + 3
3			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
2:0	dqsosc_state	R	DQS/WCK Oscillator Control State Status. ■ 000 - DQSOSC_IDLE ■ 001 - DQSOSC_START: Sending MPC ■ 010 - DQSOSC_RUNTIME: Waiting for runtime passed ■ 011 - DQSOSC_GET_RESULT1: Sending first MRR ■ 100 - DQSOSC_WAIT1: Waiting for tMRR for sending next MRR ■ 101 - DQSOSC_GET_RESULT2: Sending second MRR ■ 110 - DQSOSC_WAIT2: Waiting for tMRR or rank gap The value 0 indicates nothing is being done for DQS Oscillator. Otherwise, DQS Oscillator is running and reflects the current state of DQSOSC. It can be used for debug only to ascertain the current state of DQSOSC controller if it is stuck to one particular state. **Value After Reset:** 0x0 **Exists:** LPDDR45_DQSOSC_EN==1 **Programming Mode:** Static

1.2.30 DQSOSCCFG0

- **Description:** DQSOSC Config Register 0
- **Size:** 32 bits
- **Offset:** 0x10308
- **Exists:** LPDDR54_DQOSC_EN_OR_MEMC_DDR5==1

This register is in block REGB_DDRC_CH0.

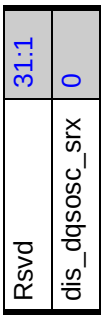


Table 1-89 Fields for Register: DQSOSCCFG0

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	dis_dqsosc_srx	R/W	■ 1 - Disable issuing of DQSOSC command sequences at Self-Refresh/SR-Powerdown exit. ■ 0 - Enable issuing of DQSOSC command sequences at Self-Refresh/SR-Powerdown exit. This is only present for designs supporting LPDDR4 or LPDDR5 or DDR5 devices. **Value After Reset:** 0x0 **Exists:** LPDDR54_DQOSC_EN_OR_MEMC_DDR5==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2

1.2.31 SCHED0

- **Description:** Scheduler Control Register 0
- **Size:** 32 bits
- **Offset:** 0x10380
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

31	30	29	28:17	16	15	x:8	7	6	5	4	3	2	1	0
opt_vprw_sch	dis_speculative_act	prefer_read	Rsvd	lpddr5_opt_act_timing	lpddr4_opt_act_timing	lpr_num_entries	autopre_rmw	dis_opt_ntt_by_pre	dis_opt_ntt_by_act	opt_wrcam_fill_level	rdwr_switch_policy_sel	pageclose	prefer_write	dis_opt_wrecc_collision_flush

Table 1-90 Fields for Register: SCHED0

Bits	Name	Memory Access	Description
31	opt_vprw_sch	R/W	Optimize exVPR/exVPW scheduling. ■ 0 - When any exVPR/exVPW are pending on CAM, read/write command of all other traffic class are masked to be scheduled ■ 1 - When any exVPR/exVPW are pending as page-hit, read/write command of all other traffic class are masked to be scheduled Program to 1 can improve utilization as other traffic class can utilize bandwidth while pages for exVPR/exVPW are being prepared or ranks/banks for exVPR/exVPW are being refreshed, but it can delay execution of exVPR/exVPW due to delay of Activate by read/write command of other traffic class. **Value After Reset:** 0x1 **Exists:** UMCTL2_VPRW_EN==1 && MEMC_ENH_RDWR_SWITCH==1 **Volatile:** true **Programming Mode:** Static

Bits	Name	Memory Access	Description
30	dis_speculative_act	R/W	Disable speculative Activate. In enhanced read write switching mode, activate commands can be issued to the other direction speculatively. This may have side-effect that the page opened for RD/WR proactively may be required to be closed to serve WR/RD respectively and it can have negative impact on performance due to command bus congestion.This register can limit such a speculative activate for the other direction. ■ 0 Allow speculative activates (default) ■ 1 Limit the speculative activates This register is effective only DDR4, LPDDR4 and LPDDR5. In this version, the value 1 is not fully verified hence this register must be set to 0. **Value After Reset:** 0x0 **Exists:** MEMC_ENH_RDWR_SWITCH==1 **Volatile:** true **Programming Mode:** Static
29	prefer_read	R/W	Make RD preferred when RD and WR have same critical level. For better RD ltancy, it is recommended to set this register to 1 but there could be efficiency drop as a trade-off with latency. ■ 0 - Disable (Default) ■ 1 - Enable (RD is preferred when same critical level between RD and WR) This register is effective only LPDDR4 and LPDDR5. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Static
28:17			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
16	lpddr5_opt_act_timing	R/W	Optimized ACT timing control for LPDDR5. This register is to be used for debug purpose. In LPDDR5, ACTIVATE command is composed of two commands, ACT-1 and ACT-2. When this register is set, ACT-1 can be issued "tRRD-2" cycle after previous ACT-2. If ACT-1 is issued at this timing, the controller does not issue ACT-2 at next cycle due to tRRD. ■ 0 - Disable (only for debug purpose) ■ 1 - Enable (Default) This register is ignored when MSTR0.lpddr5==0. This register field is only applicable for LPDDR5 mode. **Value After Reset:** 0x1 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Static
15	lpddr4_opt_act_timing	R/W	Optimized ACT timing control for LPDDR4. In LPDDR4, RD/WR/ACT takes 4 cycle. To stream Read/Write, there are only 4 cycle space between Reads/Writes. If ACT is scheduled-out after RD/WR with 1, 2 or 3 cycle gap, next RD/WR may be pushed by 1, 2 or 3 cycle and create a gap on DQ. When this register is set, ACT is not scheduled-out with the gap = 1, 2 and 3 cycle. If enabled, there could be performance impact especially for random traffic. (Latency/Utilization) ■ 1 - Enable this feature ■ 0 - Disable this feature This register is ignored when MSTR0.lpddr4==0. This register field is only applicable for LPDDR4 mode. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Static

Bits	Name	Memory Access	Description
x:8	lpr_num_entries	R/W	Number of entries in the low priority transaction store is this value + 1. (MEMC_NO_OF_ENTRY - (SCHED.lpr_num_entries + 1)) is the number of entries available for the high priority transaction store. Setting this to maximum value allocates all entries to low priority transaction store. Setting this to 0 allocates 1 entry to low priority transaction store and the rest to high priority transaction store. **Note:**In ECC configurations, the numbers of write and low priority read credits issued is one less than in the non-ECC case. One entry each is reserved in the write and low-priority read CAMs for storing the RMW requests arising out of single bit error correction RMW operation. **Value After Reset:** "MEMC_NO_OF_ENTRY/2" **Exists:** Always **Volatile:** true **Programming Mode:** Static **Range Variable[x]:** "MEMC_RDCMD_ENTRY_BITS" + 7
7	autopre_rmw	R/W	Select behavior of hif_cmd_autopre if a RMW is received on HIF with hif_cmd_autopre=1 ■ 1: Apply Autopre only for write part of RMW ■ 0: Apply Autopre for both read and write parts of RMW **Value After Reset:** 0x0 **Exists:** MEMC_USE_RMW==1 **Volatile:** true **Programming Mode:** Static
6	dis_opt_ntt_by_pre	R/W	Disable optimized NTT update by Precharge command. This register is debug purpose only. For normal operation, This register must be set to 0. ■ 1: disabled ■ 0: enabled **Value After Reset:** 0x0 **Exists:** MEMC_NTT_UPD_PRE==1 **Volatile:** true **Programming Mode:** Static

Bits	Name	Memory Access	Description
5	dis_opt_ntt_by_act	R/W	Disable optimized NTT update by Activate command. This register is debug purpose only. For normal operation, This register must be set to 0. ■ 1: disabled ■ 0: enabled **Value After Reset:** 0x0 **Exists:** MEMC_NTT_UPD_ACT==1 **Volatile:** true **Programming Mode:** Static
4	opt_wrcam_fill_level	R/W	Enable the feature of optimized write CAM fill level by switching to write when write CAM reaches certain fill level set in SCHED3.wrcam_highthresh. ■ 1: enabled ■ 0: disabled If MEMC_RDWR_SWITCH_POL_SEL==1 && rdwr_switch_policy_sel==0, this register will be ignored. It is recommended that w_max_starve is programmed as >0 value when opt_wrcam_fill_level=1 to avoid starving extremely. **Value After Reset:** 0x1 **Exists:** MEMC_ENH_RDWR_SWITCH==1 **Volatile:** true **Programming Mode:** Static
3	rdwr_switch_policy_sel	R/W	Select read write switching policy. ■ 1: select "enhanced" read write switching policy ■ 0: select "original" read write switching policy For DDR5, only "enhanced" read write switching policy is supported. **Value After Reset:** 0x1 **Exists:** MEMC_RDWR_SWITCH_POL_SEL==1 **Volatile:** true **Programming Mode:** Static

Bits	Name	Memory Access	Description
2	pageclose	R/W	If true, bank is kept open only while there are page hit transactions available in the CAM to that bank. The last read or write command in the CAM with a bank and page hit will be executed with auto-precharge if SCHEDTMG0.pageclose_timer=0. Even if this register set to 1 and SCHEDTMG0.pageclose_timer is set to 0, explicit precharge (and not auto-precharge) may be issued in some cases where there is a mode switch between Write and Read or between LPR and HPR. The Read and Write commands that are executed as part of the ECC scrub requests are also executed without auto-precharge. If false, the bank remains open until there is a need to close it (to open a different page, or for page timeout or refresh timeout) - also known as open page policy. The open page policy can be overridden by setting the per-command-autopre bit on the HIF interface (hif_cmd_autopre). The pageclose feature provides a midway between Open and Close page policies. FOR PERFORMANCE ONLY. **Value After Reset:** 0x1 **Exists:** Always **Volatile:** true **Programming Mode:** Static
1	prefer_write	R/W	If set then the bank selector prefers writes over reads. FOR DEBUG ONLY. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
0	dis_opt_wrecc_collision_flush	R/W	In this release, this register bit is required to set to 1 in software unless otherwise advised by Synopsys. **Value After Reset:** 0x1 **Exists:** MEMC_INLINE_ECC==1 **Volatile:** true **Programming Mode:** Static

1.2.32 SCHED1

- **Description:** Scheduler Control Register 1
- **Size:** 32 bits
- **Offset:** 0x10384
- **Exists:** MEMC_ENH_CAM_PTR==1

This register is in block REGB_DDRC_CH0.

31	30:28	27	26:24	23	22:20	19	18:16	15:12	11:0
opt_hit_gt_hpr	page_hit_limit_rd	Rsvd	page_hit_limit_wr	Rsvd	visible_window_limit_rd	Rsvd	visible_window_limit_wr	delay_switch_write	Rsvd

Table 1-91 Fields for Register: SCHED1

Bits	Name	Memory Access	Description
31	opt_hit_gt_hpr	R/W	Optimize the priority between Page-hit LPR and Page-miss HPR ■ 0 - Page-miss HPR has priority (default) ■ 1 - Page-hit LPR has priority This is to choose trade-off between HPR latency and total utilization. If set to 0, HPR latency can be better than 1 because HPR has priority over LPR. If set to 1, DRAM utilization can be better than 0 because number of ACT-PRE is reduced. When this register is set to 1, recommend to enable page-hit limiter so that once page-hit limiter is expired, HPR can have priority. **Value After Reset:** 0x0 **Exists:** MEMC_ENH_CAM_PTR==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
30:28	page_hit_limit_rd	R/W	Page-Hit limiter for read. When certain number of read commands are scheduled out without ACT for a bank (schedule page-hit commands), all entries belonging to the bank priority are increased equal to page-hit entry even if these are page-miss so that oldest entry belonging to the bank can be served regardless of page-hit/page-miss. The priority is reset once any ACT/PRE/AP is served to the bank. ■ 0 - Disable this feature ■ 1 - 4 commands ■ 2 - 8 commands ■ 3 - 16 commands ■ 4 - 32 commands ■ else reserved **Value After Reset:** 0x0 **Exists:** MEMC_ENH_CAM_PTR==1 **Volatile:** true **Programming Mode:** Static
27			**Reserved Field:** Yes
26:24	page_hit_limit_wr	R/W	Page-Hit limiter for write. When certain number of write commands are scheduled out without ACT for a bank (schedule page-hit commands), all entries belonging to the bank priority are increased equal to page-hit entry even if these are page-miss so that oldest entry belonging to the bank can be served regardless of page-hit/page-miss. The priority is reset once any ACT/PRE/AP is served to the bank. ■ 0 - Disable this feature ■ 1 - 4 commands ■ 2 - 8 commands ■ 3 - 16 commands ■ 4 - 32 commands ■ else reserved **Value After Reset:** 0x0 **Exists:** MEMC_ENH_CAM_PTR==1 **Volatile:** true **Programming Mode:** Static
23			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
22:20	visible_window_limit_rd	R/W	Visible window limiter for read. This is to prevent extreme starvation against other entries within a CAM. Each read CAM entry has a counter, it is set to the value programmed in this register when command is pushed and is counted-down when newer read CAM entry is scheduled-out. The counter represent starvation within RD CAM in terms of number of commands to be over taken. When the counter reaches to 0, the entry becomes expired-VPR to eliminate more starvation. ■ 0 - Disable this feature ■ 1 - 31 commands ■ 2 - 63 commands ■ 3 - 127 commands ■ 4 - 255 commands ■ else reserved **Value After Reset:** 0x0 **Exists:** MEMC_ENH_CAM_PTR==1 && UMCTL2_VPR_EN==1 **Volatile:** true **Programming Mode:** Static
19			**Reserved Field:** Yes
18:16	visible_window_limit_wr	R/W	Visible window limiter for write. This is to prevent extreme starvation against other entries within a CAM. Each write CAM entry has a counter, it is set to the value programmed in this register when command is pushed and is counted-down when newer write CAM entry is scheduled-out. The counter represent starvation within WR CAM in terms of number of commands to be over taken. When the counter reaches to 0, the entry becomes expired-VPW to eliminate more starvation. ■ 0 - Disable this feature ■ 1 - 31 commands ■ 2 - 63 commands ■ 3 - 127 commands ■ 4 - 255 commands ■ else reserved **Value After Reset:** 0x0 **Exists:** MEMC_ENH_CAM_PTR==1 && UMCTL2_VPW_EN==1 **Volatile:** true **Programming Mode:** Static

Bits	Name	Memory Access	Description
15:12	delay_switch_write	R/W	delay_switch_write indicates number of cycles to delay switching read to write mode when write page-hit request is there and no read page-hit request is there. Setting higher value may reduce number of read to write switching but increase read to write turn-around time. The register indicates the number of cycles: ■ 0: no delay ■ 1: 2 cycles delay ■ 2: 4 cycles delay ■ 3: 6 cycles delay ■ 4: 8 cycles delay ■ ... ■ 15:30 cycles delay If MEMC_RD-WR_SWITCH_POL_SEL==1 && rdwr_switch_poli-cy_sel==0, this register will be ignored. Unit: DFI clock **Value After Reset:** 0x2 **Exists:** MEMC_ENH_RDWR_SWITCH==1 **Volatile:** true **Programming Mode:** Static
11:0			**Reserved Field:** Yes

1.2.33 SCHED3

- **Description:** Scheduler Control Register 3
- **Size:** 32 bits
- **Offset:** 0x1038c
- **Exists:** MEMC_ENH_RDWR_SWITCH==1

This register is in block REGB_DDRC_CH0.

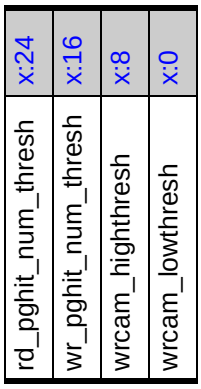


Table 1-92 Fields for Register: SCHED3

Bits	Name	Memory Access	Description
x:24	rd_pghit_num_thresh	R/W	Switch to read mode once number of read page-hit request exceeds the threshold set in the register during waiting tW2R. Set to 0 will disable the feature. If MEMC_RDWR_SWITCH_POL_SEL==1 && rdwr_switch_policy_sel==0, this register will be ignored. **Value After Reset:** 0x4 **Exists:** MEMC_ENH_RDWR_SWITCH==1 **Volatile:** true **Programming Mode:** Static **Range Variable[x]:** "MEMC_RDCMD_ENTRY_BITS" + 23
x:16	wr_pghit_num_thresh	R/W	Switch to write mode once number of write page-hit request exceeds threshold set in this register during waiting delay_switch_write timeout. Set to 0 will disable the feature. If MEMC_RDWR_SWITCH_POL_SEL==1 && rdwr_switch_policy_sel==0, this register will be ignored. **Value After Reset:** 0x4 **Exists:** MEMC_ENH_RDWR_SWITCH==1 **Volatile:** true **Programming Mode:** Static **Range Variable[x]:** "MEMC_WRCMD_ENTRY_BITS" + 15

Bits	Name	Memory Access	Description
x:8	wrcam_highthresh	R/W	The high threshold used in optimized write CAM fill level. When (MEMC_NO_OF_ENTRY - (number of loaded entries) < wrcam_highthresh), switch to write mode and prepare banks for write direction if no Exp-VPR or read collision is there. wrcam_highthresh must be set to a smaller value than wrcam_lowthresh. This feature is enabled when opt_wrcam_fill_level is 1. **Value After Reset:** 0x2 **Exists:** MEMC_ENH_RDWR_SWITCH==1 **Volatile:** true **Programming Mode:** Static **Range Variable[x]:** "MEMC_WRCMD_ENTRY_BITS" + 7
x:0	wrcam_lowthresh	R/W	The low threshold used in optimized write CAM fill level. When (MEMC_NO_OF_ENTRY - (number of loaded entries) < wrcam_lowthresh), keep to write mode and stop to prepare banks for read direction if no Exp-VPR or read collision is there. This feature is enabled when opt_wrcam_fill_level is 1. **Value After Reset:** 0x8 **Exists:** MEMC_ENH_RDWR_SWITCH==1 **Volatile:** true **Programming Mode:** Static **Range Variable[x]:** "MEMC_WRCMD_ENTRY_BITS" - 1

1.2.34 SCHED4

- **Description:** Scheduler Control Register 4
- **Size:** 32 bits
- **Offset:** 0x10390
- **Exists:** MEMC_ENH_RDWR_SWITCH==1

This register is in block REGB_DDRC_CH0.

31:24	23:16	15:8	7:0
wr_page_exp_cycles	rd_page_exp_cycles	wr_act_idle_gap	rd_act_idle_gap

Table 1-93 Fields for Register: SCHED4

Bits	Name	Memory Access	Description
31:24	wr_page_exp_cycles	R/W	wr_page_exp_cycles indicates number of cycles to keep the bank opened for write direction in read mode when both directions has request to the bank. The register indicates the number of cycle. If MEMC_RDWR_SWITCH_POL_SEL==1 && rdwr_switch_policy_sel==0, this register will be ignored. **Value After Reset:** 0x8 **Exists:** MEMC_ENH_RDWR_SWITCH==1 **Volatile:** true **Programming Mode:** Static
23:16	rd_page_exp_cycles	R/W	rd_page_exp_cycles indicates number of cycles to keep the bank opened for read direction in write mode when both directions has request to the bank. The register indicates the number of cycle. If MEMC_RDWR_SWITCH_POL_SEL==1 && rdwr_switch_policy_sel==0, this register will be ignored. **Value After Reset:** 0x40 **Exists:** MEMC_ENH_RDWR_SWITCH==1 **Volatile:** true **Programming Mode:** Static

Bits	Name	Memory Access	Description
15:8	wr_act_idle_gap	R/W	wr_act_idle_gap indicates number of cycles when write direction has no request to start preparing bank for read direction. The register indicates the number of cycle. If MEMC_RDWR_SWITCH_POL_SEL==1 && rdwr_switch_policy_sel==0, this register will be ignored. **Value After Reset:** 0x8 **Exists:** MEMC_ENH_RDWR_SWITCH==1 **Volatile:** true **Programming Mode:** Static
7:0	rd_act_idle_gap	R/W	rd_act_idle_gap indicates number of cycles when read direction has no request to start preparing bank for write direction. The register indicates the number of cycle. If MEMC_RDWR_SWITCH_POL_SEL==1 && rdwr_switch_policy_sel==0, this register will be ignored. **Value After Reset:** 0x10 **Exists:** MEMC_ENH_RDWR_SWITCH==1 **Volatile:** true **Programming Mode:** Static

1.2.35 SCHED5

- **Description:** Scheduler Control Register 5.
- **Size:** 32 bits
- **Offset:** 0x10394
- **Exists:** MEMC_ENH_RDWR_SWITCH==1 && MEMC_INLINE_ECC==1

This register is in block REGB_DDRC_CH0.

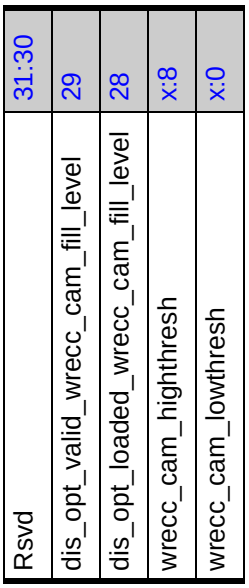


Table 1-94 Fields for Register: SCHED5

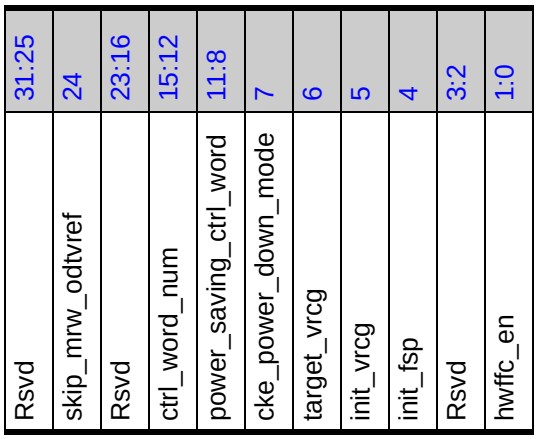
Bits	Name	Memory Access	Description
31:30			**Reserved Field:** Yes
29	dis_opt_valid_wrecc_cam_fill_level	R/W	In this release, this register bit, dis_opt_valid_wrecc_cam_fill_level, is required to set to 0 in software unless otherwise advised by Synopsys. **Value After Reset:** 0x0 **Exists:** MEMC_ENH_RDWR_SWITCH==1 && MEMC_INLINE_ECC==1 **Volatile:** true **Programming Mode:** Static

Bits	Name	Memory Access	Description
28	dis_opt_loaded_wrecc_cam_fill_level	R/W	In this release, this register bit, dis_opt_loaded_wrecc_cam_fill_level, is required to set to 0 in software unless otherwise advised by Synopsys. **Value After Reset:** 0x1 **Exists:** MEMC_ENH_RDWR_SWITCH==1 && MEMC_INLINE_ECC==1 **Volatile:** true **Programming Mode:** Static
x:8	wrecc_cam_highthresh	R/W	The high threshold used in optimized write ECC CAM fill level. When (MEMC_NO_OF_ENTRY/2 - (number of loaded entries) < wrecc_cam_highthresh), switch to write mode and prepare banks for write direction if no Exp-VPR or read collision is there. This feature is enabled when opt_wrcam_fill_level is 1. **Value After Reset:** 0x2 **Exists:** MEMC_ENH_RDWR_SWITCH==1 && MEMC_INLINE_ECC==1 **Volatile:** true **Programming Mode:** Static **Range Variable[x]:** "MEMC_WRCMD_ENTRY_BITS - 1" + 7
x:0	wrecc_cam_lowthresh	R/W	The low threshold used in optimize write ECC CAM fill level. When (MEMC_NO_OF_ENTRY/2 - (number of loaded entries) < wrecc_cam_lowthresh), keep to write mode and stop to prepare banks for read direction if no Exp-VPR or read collision is there. This feature is enabled when opt_wrcam_fill_level is 1. **Value After Reset:** 0x4 **Exists:** MEMC_ENH_RDWR_SWITCH==1 && MEMC_INLINE_ECC==1 **Volatile:** true **Programming Mode:** Static **Range Variable[x]:** "MEMC_WRCMD_ENTRY_BITS - 1" - 1

1.2.36 HWFFCCTL

- **Description:** Hardware Fast Frequency Change (HWFFC) Control Register.
- **Size:** 32 bits
- **Offset:** 0x10400
- **Exists:** UMCTL2_HWFFC_EN==1

This register is in block REGB_DDRC_CH0.



31:25	24	23:16	15:12	11:8	7	6	5	4	3:2	1:0
Rsvd	skip_mrw_odtvref	Rsvd	ctrl_word_num	power_saving_ctrl_word	cke_power_down_mode	target_vrcg	init_vrcg	init_fsp	Rsvd	hwffc_en

Table 1-95 Fields for Register: HWFFCCTL

Bits	Name	Memory Access	Description
31:25			**Reserved Field:** Yes
24	skip_mrw_odtvref	R/W	If this register is set to 1, the DDRCTL will skip issuing MRW to MR11, MR12, MR14, and MR22 as part of the LPDDR4 HWFFC procedure. In LPDDR4X SDRAM, these registers are programmed per-rank during initialization. **Value After Reset:** 0x0 **Exists:** UMCTL2_HWFFC_EN==1 && DDRCTL_LPDDR **Programming Mode:** Static
23:16			**Reserved Field:** Yes
15:12	ctrl_word_num	R/W	Number of control words must be issued to RCD while DDR4 HWFFC sequence is working.If user set this register to 0, controller does not issue MR7. DDR5: Not supported. **Value After Reset:** 0x0 **Exists:** UMCTL2_HWFFC_EN==1 && DDRCTL_DDR **Programming Mode:** Static

Bits	Name	Memory Access	Description
11:8	power_saving_ctrl_word	R/W	Indicates the value to be loaded into power saving setting control word (F0RC09). User need to set this register when change frequency using sequence A. Used in DDR4 RDIMM. DDR5: Not supported. **Value After Reset:** 0x0 **Exists:** UMCTL2_HWFFC_EN==1 && DDRCTL_DDR **Programming Mode:** Static
7	cke_power_down_mode	R/W	Set to 1 when the DDRCTL issues MR7 for F0RC09 while DDR4 HWFFC sequence is working if necessary. Used in DDR4 RDIMM or LRDIMM. DDR5: Not supported. **Value After Reset:** 0x0 **Exists:** UMCTL2_HWFFC_EN==1 && DDRCTL_DDR **Programming Mode:** Static
6	target_vrcg	R/W	LPDDR4: Set target value of VRCG (MR13 OP[3]). DDR4: Set this to 1 if CKE Power down mode in the RCD Power Saving Settings Control Word (F0RC09) needs to be enabled after clock frequency is changed as part of HWFFC procedure. DDR5: Not supported. **Value After Reset:** 0x0 **Exists:** UMCTL2_HWFFC_EN==1 **Programming Mode:** Static
5	init_vrcg	R/W	LPDDR4: Set initial value of VRCG (MR13 OP[3]). This field value is used when HWFFCCTL.hwffc_en has been changed to 2'b11. DDR4: Set this to 1 if CKE Power down mode in the RCD Power Saving Settings Control Word (F0RC09) needs to be disabled before clock frequency is changed as part of HWFFC procedure. DDR5: Not supported. **Value After Reset:** 0x0 **Exists:** UMCTL2_HWFFC_EN==1 **Programming Mode:** Static
4	init_fsp	R/W	Set initial value of FSP-OP (MR13 OP[7]). This field value is used when HWFFCCTL.hwffc_en has been changed to 2'b11. **Value After Reset:** 0x1 **Exists:** UMCTL2_HWFFC_EN==1 && DDRCTL_LPDDR==1 **Programming Mode:** Static
3:2			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
1:0	hwffc_en	R/W	Enable HWFFC through Hardware Low Power Interface. The other fields of this register is used only when changing this field to 2'b11. ■ 00 - Disable HWFFC ■ 10 - Intermediate, set only when disabling HWFFC ■ 11 - Enable HWFFC ■ 01 - Not allowed DDR5: Not supported. **Value After Reset:** 0x0 **Exists:** UMCTL2_HWFFC_EN==1 **Programming Mode:** Dynamic

1.2.37 HWFFCSTAT

- **Description:** Hardware Fast Frequency Change (HWFFC) Status Register
- **Size:** 32 bits
- **Offset:** 0x10404
- **Exists:** UMCTL2_HWFFC_EN==1

This register is in block REGB_DDRC_CH0.

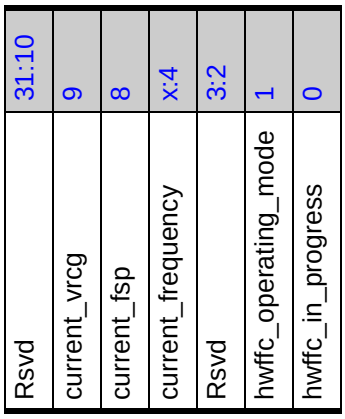


Table 1-96 Fields for Register: HWFFCSTAT

Bits	Name	Memory Access	Description
31:10			**Reserved Field:** Yes
9	current_vrcg	R	Indicates current value of VRCG (MR13 OP[3]). **Value After Reset:** 0x1 **Exists:** UMCTL2_HWFFC_EN==1 **Programming Mode:** Dynamic
8	current_fsp	R	Indicates current value of FSP-OP (MR13 OP[7]). **Value After Reset:** 0x1 **Exists:** UMCTL2_HWFFC_EN==1 && DDRCTL_LPDDR==1 **Programming Mode:** Dynamic
x:4	current_frequency	R	Indicates the current frequency. ■ 0 - Frequency 0/Normal ■ 1 - Frequency 1/FREQ1 ■ 2 - Frequency 2/FREQ2 ■ 3 - Frequency 3/FREQ3 **Value After Reset:** 0x0 **Exists:** UMCTL2_HWFFC_EN==1 **Programming Mode:** Dynamic **Range Variable[x]:** "DDRCTL_FREQUENCY_BITS" + 3

Bits	Name	Memory Access	Description
3:2			**Reserved Field:** Yes
1	hwffc_operating_mode	R	Operating mode of HWFFC. ■ 0 - Normal ■ 1 - Self Refresh or SR-Powerdown **Value After Reset:** 0x0 **Exists:** UMCTL2_HWFFC_EN==1 **Programming Mode:** Dynamic
0	hwffc_in_progress	R	Indicates HWFFC is in progress. **Value After Reset:** 0x0 **Exists:** UMCTL2_HWFFC_EN==1 **Programming Mode:** Dynamic

1.2.38 DFILPCFG0

- **Description:** DFI Low Power Configuration Register 0
- **Size:** 32 bits
- **Offset:** 0x10500
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

31:21	20	19:17	16	15:9	8	7:5	4	3:1	0
Rsvd	dfi_lp_data_req_en	Rsvd	dfi_lp_en_data	Rsvd	dfi_lp_en_dsm	Rsvd	dfi_lp_en_sr	Rsvd	dfi_lp_en_pd

Table 1-97 Fields for Register: DFILPCFG0

Bits	Name	Memory Access	Description
31:21			**Reserved Field:** Yes
20	dfi_lp_data_req_en	R/W	Enables DFI Data Low Power interface. ■ 0 - Disabled. dfi_lp_data_req is not asserted. ■ 1 - Enabled **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Static
19:17			**Reserved Field:** Yes
16	dfi_lp_en_data	R/W	Enables DFI Data Low Power interface handshaking during data bus idle. ■ 0 - Disabled ■ 1 - Enabled **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
15:9			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
8	dfi_lp_en_dsm	R/W	Enables DFI Low Power interface handshaking during Deep Sleep Mode Entry/Exit. ■ 0 - Disabled ■ 1 - Enabled This is only present for designs supporting LPDDR5 devices. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Static
7:5			**Reserved Field:** Yes
4	dfi_lp_en_sr	R/W	Enables DFI Low Power interface handshaking during Self Refresh Entry/Exit. ■ 0 - Disabled ■ 1 - Enabled **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
3:1			**Reserved Field:** Yes
0	dfi_lp_en_pd	R/W	Enables DFI Low Power interface handshaking during Power Down Entry/Exit. ■ 0 - Disabled ■ 1 - Enabled **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.2.39 DFIUPD0

- **Description:** DFI Update Register 0
- **Size:** 32 bits
- **Offset:** 0x10508
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

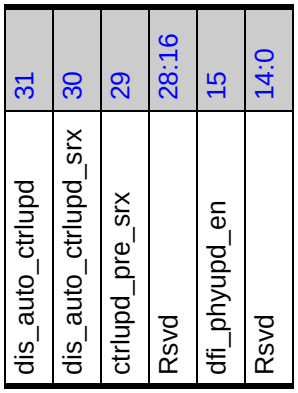


Table 1-98 Fields for Register: DFIUPD0

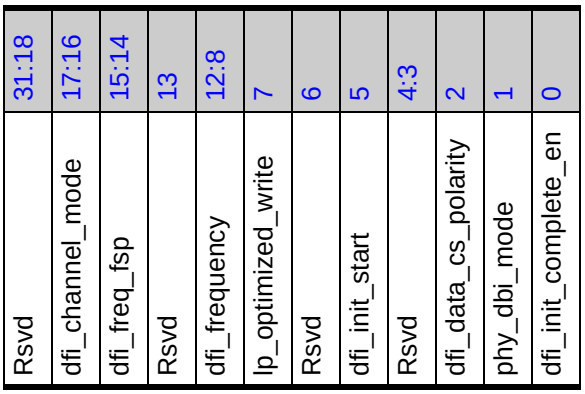
Bits	Name	Memory Access	Description
31	dis_auto_ctrlupd	R/W	■ 0 - DDRCTL issues dfi_ctrlupd_req periodically. ■ 1 - Disable the automatic dfi_ctrlupd_req generation by the DDRCTL. The core must issue the dfi_ctrlupd_req signal using register OPCTRLCMD.ctrlupd. Don't care for DDR5. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
30	dis_auto_ctrlupd_srx	R/W	■ 0 - DDRCTL issues a dfi_ctrlupd_req before or after exiting self-refresh, depending on DFIUPD0.ctrlupd_pre_srx. ■ 1 - Disable the automatic dfi_ctrlupd_req generation by the DDRCTL at self-refresh exit. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Static

Bits	Name	Memory Access	Description
29	ctrlupd_pre_srx	R/W	Selects dfi_ctrlupd_req requirements at SRX: ■ 0 : send ctrlupd after SRX ■ 1 : send ctrlupd before SRX If DFIUPD0.dis_auto_ctrlupd_srx=1, this register has no impact, because no dfi_ctrlupd_req will be issued when SRX. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
28:16			**Reserved Field:** Yes
15	dfi_phyupd_en	R/W	Enables the support for acknowledging PHY-initiated updates: ■ 0 - Disabled ■ 1 - Enabled **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Static
14:0			**Reserved Field:** Yes

1.2.40 DFIMISC

- **Description:** DFI Miscellaneous Control Register
- **Size:** 32 bits
- **Offset:** 0x10510
- **Exists:** Always

This register is in block REGB_DDRC_CH0.



31:18	17:16	15:14	13	12:8	7	6	5	4:3	2	1	0
Rsvd	dfi_channel_mode	dfi_freq_fsp	Rsvd	dfi_frequency	lp_optimized_write	Rsvd	dfi_init_start	Rsvd	dfi_data_cs_polarity	phy_dbi_mode	dfi_init_complete_en

Table 1-99 Fields for Register: DFIMISC

Bits	Name	Memory Access	Description
31:18			**Reserved Field:** Yes
17:16	dfi_channel_mode	R/W	This field controls how internal DFI data is connected to dfi0_*data* and dfi1_*data*. Under the following conditions, this must be set to 2'b01: ■ When using a Synopsys DWC LPDDR54 PHY ■ Single DDRC Dual DFI configuration (MEMC_DRAM_-DATA_WIDTH=32) ■ Data width of each DFI channel is 16 ■ Each DFI channel corresponds to 16-bit LPDDR5/4 channel (32-bit in total) Otherwise, this must be set to 2'b00. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Static
15:14	dfi_freq_fsp	R/W	This register value propagates to dfi_freq_fsp pin directly. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic
13			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
12:8	dfi_frequency	R/W	Indicates the operating frequency of the system. The number of supported frequencies and the mapping of signal values to clock frequencies are defined by the PHY. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1
7	lp_optimized_write	R/W	If this bit is 1, LPDDR4 write DQ is set to 8'hF8 if masked write with enabling DBI; otherwise, that value is set to 8'hFF **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
6			**Reserved Field:** Yes
5	dfi_init_start	R/W	PHY init start request signal.When asserted it triggers the PHY init start request **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
4:3			**Reserved Field:** Yes
2	dfi_data_cs_polarity	R/W	Defines polarity of dfi_wrdata_cs and dfi_rddata_cs signals. ■ 0: Signals are active low ■ 1: Signals are active high **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
1	phy_dbi_mode	R/W	DBI implemented in DDRC or PHY. ■ 0 - DDRC implements DBI functionality. ■ 1 - PHY implements DBI functionality. Present only in designs configured to support DDR4 and LPDDR4. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

Bits	Name	Memory Access	Description
0	dfi_init_complete_en	R/W	PHY initialization complete enable signal. When asserted the dfi_init_complete signal can be used to trigger SDRAM initialization **Value After Reset:** 0x1 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3

1.2.41 DFISTAT

- **Description:** DFI Status Register
- **Size:** 32 bits
- **Offset:** 0x10514
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

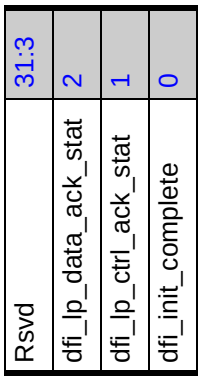


Table 1-100 Fields for Register: DFISTAT

Bits	Name	Memory Access	Description
31:3			**Reserved Field:** Yes
2	dfi_lp_data_ack_stat	R	Stores the value of the dfi_lp_data_ack input to the controller. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
1	dfi_lp_ctrl_ack_stat	R	Stores the value of the dfi_lp_ctrl_ack input to the controller. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
0	dfi_init_complete	R	The status flag register which announces when the DFI initialization has been completed. The DFI INIT triggered by dfi_init_start signal and then the dfi_init_complete flag is polled to know when the initialization is done. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.2.42 DFIPHYMSTR

- **Description:** DFI PHY Master
- **Size:** 32 bits
- **Offset:** 0x10518
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

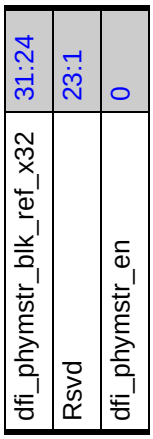


Table 1-101 Fields for Register: DFIPHYMSTR

Bits	Name	Memory Access	Description
31:24	dfi_phymstr_blk_ref_x32	R/W	The programmed value x32 is the maximum number of DFI clock cycles that allows to send pending refreshes before starting self-refresh entry process. - 0x00 - 0 DFI clock cycles, no delay - 0x01 - 32 DFI clock cycles - ... - 0xFF - 8160 DFI clock cycles Unit: Multiples of 32 DFI clock cycles. Note: Use as default value (0x80) unless Synopsys suggest to change value. **Value After Reset:** 0x80 **Exists:** Always **Programming Mode:** Static
23:1			**Reserved Field:** Yes
0	dfi_phymstr_en	R/W	Enables the PHY Master Interface: ■ 0 - Disabled ■ 1 - Enabled **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Static

1.2.43 DFI0MSGCTL0

- **Description:** DFI0 Message Control Register 0.
- **Size:** 32 bits
- **Offset:** 0x10520
- **Exists:** DDRCTL_DFI_CTRLMSG==1

This register is in block REGB_DDRC_CH0.

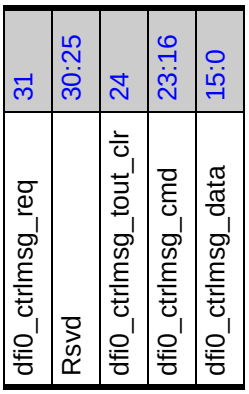


Table 1-102 Fields for Register: DFI0MSGCTL0

Bits	Name	Memory Access	Description
31	dfi0_ctrlmsg_req	R/W1S	Setting this register bit to 1 triggers a DFI controller message transmission operation. DDRCTL automatically clear this bit when the DFI controller message request (dfi0_ctrlmsg_req) is asserted at the DFI MC to PHY Message port interface . This bit must be programmed separately after programming other register fields appropriately of this register. Note: ■ DFI controller message request can be issued only if DFIPHYMSTR.dfi_phymstr_en = 1 ■ DFI controller message request must not be set during DFI LP mode due to software controlled low power entry. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
30:25			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
24	dfi0_ctrlmsg_tout_clr	R/W1C	If this bit is set, DFI0MSGSTAT0.dfi0_ctrlmsg_resp_tout is cleared by the controller. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
23:16	dfi0_ctrlmsg_cmd	R/W	DFI0 controller message command. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
15:0	dfi0_ctrlmsg_data	R/W	DFI0 controller message data. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.2.44 DFI0MSGSTAT0

- **Description:** DFI0 Message Status Register 0
- **Size:** 32 bits
- **Offset:** 0x10524
- **Exists:** DDRCTL_DFI_CTRLMSG==1

This register is in block REGB_DDRC_CH0.

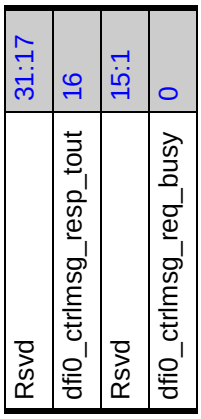


Table 1-103 Fields for Register: DFI0MSGSTAT0

Bits	Name	Memory Access	Description
31:17			**Reserved Field:** Yes
16	dfi0_ctrlmsg_resp_tout	R	This bit is set if dfi0_ctrlmsg_ack is not asserted by PHY within dfi_t_ctrlmsg_resp after asserting dfi0_ctrlmsg_req **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
15:1			**Reserved Field:** Yes
0	dfi0_ctrlmsg_req_busy	R	The SoC must trigger DFI controller message request only if this signal is low. This signal goes high in the clock after the DDRCTL accepts software triggered DFI controller message request by writing into DFI0MSGCTRL0.dfi0_ctrlmsg_req. It goes low when PHY deasserts dfi0_ctrlmsg_ack or dfi0_ctrlmsg_resp_tout event has triggered. ■ 0 - Indicates that the SoC core can initiate a DFI controller message request operation ■ 1 - Indicates that DFI controller message request operation is in progress **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.2.45 POISONCFG

- **Description:** AXI Poison Configuration Register. Common for all AXI ports
- **Size:** 32 bits
- **Offset:** 0x10580
- **Exists:** UMCTL2_INCL_ARB==1 && UMCTL2_A_AXI==1

This register is in block REGB_DDRC_CH0.

31:25	24	23:21	20	19:17	16	15:9	8	7:5	4	3:1	0
Rsvd	rd_poison_intr_clr	Rsvd	rd_poison_intr_en	Rsvd	rd_poison_slverr_en	Rsvd	wr_poison_intr_clr	Rsvd	wr_poison_intr_en	Rsvd	wr_poison_slverr_en

Table 1-104 Fields for Register: POISONCFG

Bits	Name	Memory Access	Description
31:25			**Reserved Field:** Yes
24	rd_poison_intr_clr	R/W1C	Interrupt clear for read transaction poisoning. Allow 2/3 clock cycles for correct value to propagate to core logic and clear the interrupts. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
23:21			**Reserved Field:** Yes
20	rd_poison_intr_en	R/W	If set to 1, enables interrupts for read transaction poisoning **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Dynamic
19:17			**Reserved Field:** Yes
16	rd_poison_slverr_en	R/W	If set to 1, enables SLVERR response for read transaction poisoning **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
15:9			**Reserved Field:** Yes
8	wr_poison_intr_clr	R/W1C	Interrupt clear for write transaction poisoning. Allow 2/3 clock cycles for correct value to propagate to core logic and clear the interrupts. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
7:5			**Reserved Field:** Yes
4	wr_poison_intr_en	R/W	If set to 1, enables interrupts for write transaction poisoning **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Dynamic
3:1			**Reserved Field:** Yes
0	wr_poison_slverr_en	R/W	If set to 1, enables SLVERR response for write transaction poisoning **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Dynamic

1.2.46 POISONSTAT

- **Description:** AXI Poison Status Register
- **Size:** 32 bits
- **Offset:** 0x10584
- **Exists:** UMCTL2_INCL_ARB==1 && UMCTL2_A_AXI==1

This register is in block REGB_DDRC_CH0.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
rd_poison_intr_15	rd_poison_intr_14	rd_poison_intr_13	rd_poison_intr_12	rd_poison_intr_11	rd_poison_intr_10	rd_poison_intr_9	rd_poison_intr_8	rd_poison_intr_7	rd_poison_intr_6	rd_poison_intr_5	rd_poison_intr_4	rd_poison_intr_3	rd_poison_intr_2	rd_poison_intr_1	rd_poison_intr_0	wr_poison_intr_15	wr_poison_intr_14	wr_poison_intr_13	wr_poison_intr_12	wr_poison_intr_11	wr_poison_intr_10	wr_poison_intr_9	wr_poison_intr_8	wr_poison_intr_7	wr_poison_intr_6	wr_poison_intr_5	wr_poison_intr_4	wr_poison_intr_3	wr_poison_intr_2	wr_poison_intr_1	wr_poison_intr_0

Table 1-105 Fields for Register: POISONSTAT

Bits	Name	Memory Access	Description
31	rd_poison_intr_15	R	Read transaction poisoning error interrupt for port 15. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's read address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register rd_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_15==1 **Programming Mode:** Dynamic
30	rd_poison_intr_14	R	Read transaction poisoning error interrupt for port 14. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's read address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register rd_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_14==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
29	rd_poison_intr_13	R	Read transaction poisoning error interrupt for port 13. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's read address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register rd_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_13==1 **Programming Mode:** Dynamic
28	rd_poison_intr_12	R	Read transaction poisoning error interrupt for port 12. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's read address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register rd_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_12==1 **Programming Mode:** Dynamic
27	rd_poison_intr_11	R	Read transaction poisoning error interrupt for port 11. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's read address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register rd_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_11==1 **Programming Mode:** Dynamic
26	rd_poison_intr_10	R	Read transaction poisoning error interrupt for port 10. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's read address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register rd_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_10==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
25	rd_poison_intr_9	R	Read transaction poisoning error interrupt for port 9. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's read address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register rd_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_9==1 **Programming Mode:** Dynamic
24	rd_poison_intr_8	R	Read transaction poisoning error interrupt for port 8. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's read address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register rd_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_8==1 **Programming Mode:** Dynamic
23	rd_poison_intr_7	R	Read transaction poisoning error interrupt for port 7. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's read address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register rd_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_7==1 **Programming Mode:** Dynamic
22	rd_poison_intr_6	R	Read transaction poisoning error interrupt for port 6. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's read address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register rd_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_6==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
21	rd_poison_intr_5	R	Read transaction poisoning error interrupt for port 5. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's read address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register rd_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_5==1 **Programming Mode:** Dynamic
20	rd_poison_intr_4	R	Read transaction poisoning error interrupt for port 4. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's read address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register rd_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_4==1 **Programming Mode:** Dynamic
19	rd_poison_intr_3	R	Read transaction poisoning error interrupt for port 3. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's read address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register rd_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_3==1 **Programming Mode:** Dynamic
18	rd_poison_intr_2	R	Read transaction poisoning error interrupt for port 2. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's read address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register rd_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_2==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
17	rd_poison_intr_1	R	Read transaction poisoning error interrupt for port 1. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's read address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register rd_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_1==1 **Programming Mode:** Dynamic
16	rd_poison_intr_0	R	Read transaction poisoning error interrupt for port 0. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's read address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register rd_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_0==1 **Programming Mode:** Dynamic
15	wr_poison_intr_15	R	Write transaction poisoning error interrupt for port 15. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's write address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register wr_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_15==1 **Programming Mode:** Dynamic
14	wr_poison_intr_14	R	Write transaction poisoning error interrupt for port 14. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's write address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register wr_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_14==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
13	wr_poison_intr_13	R	Write transaction poisoning error interrupt for port 13. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's write address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register wr_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_13==1 **Programming Mode:** Dynamic
12	wr_poison_intr_12	R	Write transaction poisoning error interrupt for port 12. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's write address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register wr_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_12==1 **Programming Mode:** Dynamic
11	wr_poison_intr_11	R	Write transaction poisoning error interrupt for port 11. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's write address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register wr_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_11==1 **Programming Mode:** Dynamic
10	wr_poison_intr_10	R	Write transaction poisoning error interrupt for port 10. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's write address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register wr_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_10==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
9	wr_poison_intr_9	R	Write transaction poisoning error interrupt for port 9. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's write address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register wr_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_9==1 **Programming Mode:** Dynamic
8	wr_poison_intr_8	R	Write transaction poisoning error interrupt for port 8. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's write address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register wr_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_8==1 **Programming Mode:** Dynamic
7	wr_poison_intr_7	R	Write transaction poisoning error interrupt for port 7. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's write address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register wr_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_7==1 **Programming Mode:** Dynamic
6	wr_poison_intr_6	R	Write transaction poisoning error interrupt for port 6. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's write address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register wr_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_6==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
5	wr_poison_intr_5	R	Write transaction poisoning error interrupt for port 5. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's write address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register wr_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_5==1 **Programming Mode:** Dynamic
4	wr_poison_intr_4	R	Write transaction poisoning error interrupt for port 4. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's write address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register wr_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_4==1 **Programming Mode:** Dynamic
3	wr_poison_intr_3	R	Write transaction poisoning error interrupt for port 3. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's write address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register wr_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_3==1 **Programming Mode:** Dynamic
2	wr_poison_intr_2	R	Write transaction poisoning error interrupt for port 2. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's write address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register wr_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_2==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
1	wr_poison_intr_1	R	Write transaction poisoning error interrupt for port 1. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's write address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register wr_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_1==1 **Programming Mode:** Dynamic
0	wr_poison_intr_0	R	Write transaction poisoning error interrupt for port 0. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's write address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register wr_poison_intr_clr, then value propagated to APB clock. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_0==1 **Programming Mode:** Dynamic

1.2.47 ECCCFG0

- **Description:** ECC Configuration Register 0
- **Size:** 32 bits
- **Offset:** 0x10600
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH0.

31:30	29	x:24	23:22	21:16	15	14:8	7	6	5:4	3	2:0
ecc_region_map_granu	ecc_region_map_other	ecc_ap_err_threshold	Rsvd	blk_channel_idle_time_x32	Rsvd	ecc_region_map	ecc_region_remap_en	ecc_ap_en	Rsvd	test_mode	ecc_mode

Table 1-106 Fields for Register: ECCCFG0

Bits	Name	Memory Access	Description
31:30	ecc_region_map_granu	R/W	Granularity of Selectable Protected Region. Define one region size for ECCCFG0.ecc_region_map ■ 0 - 1/8 of memory spaces ■ 1 - 1/16 of memory spaces ■ 2 - 1/32 of memory spaces ■ 3 - 1/64 of memory spaces **Value After Reset:** 0x0 **Exists:** MEMC_INLINE_ECC==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
29	ecc_region_map_other	R/W	When ECCCFG0.ecc_region_map_granu>0, there is a region which is not controlled by ecc_region_map. This register defines the region to be protected or non-protected for Inline ECC. ■ 0 - Non-Protected ■ 1 - Protected This register is valid only when ECCCFG0.ecc_region_map_granu>0 && ECCCFG0.ecc_mode=4. **Value After Reset:** 0x0 **Exists:** MEMC_INLINE_ECC==1 **Programming Mode:** Static
x:24	ecc_ap_err_threshold	R/W	Set threshold for address parity error. ECCAPSTAT.ecc_ap_err is asserted if number of ECC errors (correctable/uncorrectable) within one burst exceeds this threshold. This register value must be less than "Total number of ECC checks within one burst" when this feature is used, "Total number of ECC check within one burst" is calculated by (DRAM Data width) x (DRAM BL) / 64. **Value After Reset:** "MEMC_MAX_INLINE_ECC_PER_BURST/2-1" **Exists:** MEMC_ECCAP==1 **Programming Mode:** Static **Range Variable[x]:** "MEMC_MAX_INLINE_ECC_PER_BURST_BITS" + 23
23:22			**Reserved Field:** Yes

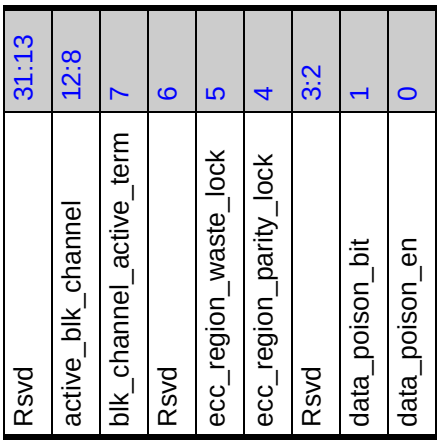
Bits	Name	Memory Access	Description
21:16	blk_channel_idle_time_x32	R/W	Indicates the number of cycles on HIF interface with no access to protected regions which will cause flush of all the block channels. In order to flush block channel, DDRCTL injects write ECC command (when there is no incoming HIF command) if there is any write in the block and then stop tracking the block address. ■ 0 indicates no timeout (feature is disabled, not supported with this version) ■ 1 indicates 32 cycles ■ 2 indicates 2*32 cycles, etc. Unit: Multiples of 32 DRAM clock cycles. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x3f **Exists:** MEMC_INLINE_ECC==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
15			**Reserved Field:** Yes
14:8	ecc_region_map	R/W	Selectable Protected Region setting. Memory space is divided to 8/16/32/64 regions which is determined by ECCCFG0.ecc_region_map_granu. Note: Highest 1/8 memory space is always ECC region. Lowest 7 regions are Selectable Protected Regions. The Selectable Protected Regions can be protected/non-protected selectively by ECCCFG0.ecc_region_map[6:0]. Other upper regions are non-protected region if any. Each bit of ECCCFG0.ecc_region_map[6:0] correspond to each of lowest 7 regions respectively. In order to protect a region with ECC, set the corresponding bit to 1, otherwise set to 0. All "0"s is invalid - there must be at least one protected region if inline ECC is enabled via ECCCFG0.ecc_mode register. All regions are protected with the following setting. ■ ecc_region_map=7'b1111111 ■ ecc_region_map_granu=0 Only first 1/64 region is protected with the following setting. ■ ecc_region_map=7'b0000001 ■ ecc_region_map_granu=3 **Value After Reset:** 0x7f **Exists:** MEMC_INLINE_ECC==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3

Bits	Name	Memory Access	Description
7	ecc_region_remap_en	R/W	Enables remapping ECC region feature. Only supported when inline ECC is enabled. ■ 0 - Disable ■ 1 - Enable **Value After Reset:** 0x0 **Exists:** MEMC_INLINE_ECC==1 **Programming Mode:** Static
6	ecc_ap_en	R/W	Enable address protection feature. Only supported when inline ECC is enabled. ■ 0: disable ■ 1: enable **Value After Reset:** 0x1 **Exists:** MEMC_ECCAP==1 **Programming Mode:** Static
5:4			**Reserved Field:** Yes
3	test_mode	R/W	If this bit is set to 1, no ECC is performed, and the ECC byte is accessed directly from co_wu_rxdata_ecc and ra_co_resp_ecc_data. This test mode is only supported with the HIF interface (DDRCTL_SYS_INTF=0). This test mode is only supported in full and half bus width mode. In other words, if MSTR0.data_bus_width is not equal to 0 or 1, this test_mode field must be set to 0. If test_mode is set to 1, the ecc_mode field must be set to 3'b000. Note: test_mode is not supported in inline ECC mode and the register value is don't care. **Value After Reset:** 0x0 **Exists:** UMCTL2_INCL_ARB == 0 && MEMC_ECC_SUPPORT>0 **Programming Mode:** Static
2:0	ecc_mode	R/W	ECC mode indicator ■ 000 - ECC disabled ■ 100 - ECC enabled - SEC/DED ■ 101 - ECC enabled - Advanced ECC (Illegal value when MEMC_INLINE_ECC=1) ■ all other settings are reserved for future use **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Static

1.2.48 ECCCFG1

- **Description:** ECC Configuration Register 1
- **Size:** 32 bits
- **Offset:** 0x10604
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH0.



31:13	12:8	7	6	5	4	3:2	1	0
Rsvd	active_blk_channel	blk_channel_active_term	Rsvd	ecc_region_waste_lock	ecc_region_parity_lock	Rsvd	data_poison_bit	data_poison_en

Table 1-107 Fields for Register: ECCCFG1

Bits	Name	Memory Access	Description
31:13			**Reserved Field:** Yes
12:8	active_blk_channel	R/W	Number of active block channels. Total number of ECC block channels are defined by MEMC_NO_OF_BLK_CHANNEL hardware parameter. This register can limit the number of available channels. For example, if set to 0, only one channel is active and therefore block interleaving is disabled. The valid range is from 0 to MEMC_NO_OF_BLK_CHANNEL-1. **Value After Reset:** "MEMC_NO_OF_BLK_CHANNEL-1" **Exists:** MEMC_INLINE_ECC==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3

Bits	Name	Memory Access	Description
7	blk_channel_active_term	R/W	Block Channel active terminate enable. If enabled, block channel is terminated when full block write or full block read is performed (all address within block are written or read) ■ 0 - Disable (only for debug purpose) ■ 1 - Enable (default) This is debug register, and this must be set to 1 for normal operation. **Value After Reset:** 0x1 **Exists:** MEMC_INLINE_ECC==1 **Volatile:** true **Programming Mode:** Static
6			**Reserved Field:** Yes
5	ecc_region_waste_lock	R/W	Locks the remaining waste parts of the ECC region (hole) that are not locked by ecc_region_parity_lock. ■ 1: Locked; if this region is accessed, error response is generated. ■ 0: Unlocked; this region can be accessed normally, similar to non-ECC protected region. **Value After Reset:** 0x1 **Exists:** MEMC_INLINE_ECC==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
4	ecc_region_parity_lock	R/W	Locks the parity section of the ECC region (hole) which is the highest system address part of the memory that stores ECC parity for protected region. ■ 1: Locked; if this region is accessed, error response is generated. ■ 0: Unlocked; this region can be accessed normally, similar to non-ECC protected region. **Value After Reset:** 0x1 **Exists:** MEMC_INLINE_ECC==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
3:2			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
1	data_poison_bit	R/W	Selects whether to poison 1 or 2 bits - if 0 -> 2-bit (uncorrectable) data poisoning, if 1 -> 1-bit (correctable) data poisoning, if ECCCFG1.data_poison_en=1. Valid only when MEMC_ECC_SUPPORT==1 or 3 (in SECDED ECC mode i.e ECCCFG0.ecc_mode=3'b100) **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
0	data_poison_en	R/W	Enable ECC data poisoning - introduces ECC errors on writes to address specified by the ECCPOISONADDR0/1 registers This field must be set to 0 if ECC is disabled (ECCCFG0.ecc_mode = 0). **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3

1.2.49 ECCSTAT

- **Description:** SECDED ECC Status Register
- **Size:** 32 bits
- **Offset:** 0x10608
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH0.
Valid only in MEMC_ECC_SUPPORT==1 or 3 (SECDED ECC mode)

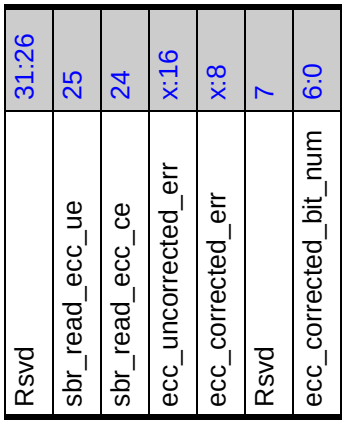


Table 1-108 Fields for Register: ECCSTAT

Bits	Name	Memory Access	Description
31:26			**Reserved Field:** Yes
25	sbr_read_ecc_ue	R	Indicates the uncorrectable error interrupt is due to read operation by scrubber. This bit is cleared on ECCCTL.ecc_uncorrected_err_clr ■ 0 - Mainline/Demand read uncorrectable error interrupt ■ 1 - Scrubber read uncorrectable error interrupt **Value After Reset:** 0x0 **Exists:** UMCTL2_SBR_EN_1==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
24	sbr_read_ecc_ce	R	Indicates the correctable error interrupt is due to read operation by scrubber. This bit is cleared on ECCCTL.ecc_corrected_err_clr ■ 0 - Mainline/Demand read correctable error interrupt ■ 1 - Scrubber read correctable error interrupt **Value After Reset:** 0x0 **Exists:** UMCTL2_SBR_EN_1==1 **Programming Mode:** Static
x:16	ecc_uncorrected_err	R	Double-bit error indicator. In sideband ECC mode, 1 bit per ECC lane. In inline ECC mode, the register always is 1 bit to indicate uncorrectable error on any lane. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Static **Range Variable[x]:** "(MEMC_INLINE_ECC_EN==1 && MEMC_SIDEBAND_ECC_EN==0) ? 1 : (MEMC_FREQ_RATIO==4) ? 8 : 4" + 15
x:8	ecc_corrected_err	R	Single-bit error indicator. In sideband ECC mode, 1 bit per ECC lane. In inline ECC mode, the register always is 1 bit to indicate correctable error on any lane. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Static **Range Variable[x]:** "(MEMC_INLINE_ECC_EN==1 && MEMC_SIDEBAND_ECC_EN==0) ? 1 : (MEMC_FREQ_RATIO==4) ? 8 : 4" + 7
7			**Reserved Field:** Yes
6:0	ecc_corrected_bit_num	R	Bit number corrected by single-bit ECC error. See ECC section of architecture chapter for encoding of this field. If more than one data lane has an error, the lower data lane is selected. This register is 7 bits wide in order to handle 72 bits of the data present in a single lane. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Static

1.2.50 ECCCTL

- **Description:** ECC Clear Register
- **Size:** 32 bits
- **Offset:** 0x1060c
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH0.

31:19	18	17	16	15:11	10	9	8	7:5	4	3	2	1	0
Rsvd	ecc_ap_err_intr_force	ecc_uncorrected_err_intr_force	ecc_corrected_err_intr_force	Rsvd	ecc_ap_err_intr_en	ecc_uncorrected_err_intr_en	ecc_corrected_err_intr_en	Rsvd	ecc_ap_err_intr_clr	ecc_uncorr_err_cnt_clr	ecc_corr_err_cnt_clr	ecc_uncorrected_err_clr	ecc_corrected_err_clr

Table 1-109 Fields for Register: ECCCTL

Bits	Name	Memory Access	Description
31:19			**Reserved Field:** Yes
18	ecc_ap_err_intr_force	R/W1C	Interrupt force bit for ecc_ap_err_intr. Setting this register will cause the output interrupt to be asserted. DDRCTL automatically clears this bit. There is no interaction between functionally triggering an interrupt and forcing an interrupt (they are mutually exclusive). **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 && MEMC_ECCAP==1 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
17	ecc_uncorrected_err_intr_force	R/W1C	Interrupt force bit for ecc_uncorrected_err_intr. Setting this register will cause the output interrupt to be asserted. DDRCTL automatically clears this bit. There is no interaction between functionally triggering an interrupt and forcing an interrupt (they are mutually exclusive). **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Testable:** readOnly **Programming Mode:** Dynamic
16	ecc_corrected_err_intr_force	R/W1C	Interrupt force bit for ecc_corrected_err_intr. Setting this register will cause the output interrupt to be asserted. DDRCTL automatically clears this bit. There is no interaction between functionally triggering an interrupt and forcing an interrupt (they are mutually exclusive). **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Testable:** readOnly **Programming Mode:** Dynamic
15:11			**Reserved Field:** Yes
10	ecc_ap_err_intr_en	R/W	Interrupt enable bit for ecc_ap_err_intr. ■ 1: Enabled ■ 0: Disabled **Value After Reset:** 0x1 **Exists:** MEMC_ECC_SUPPORT>0 && MEMC_ECCAP==1 **Programming Mode:** Dynamic
9	ecc_uncorrected_err_intr_en	R/W	Interrupt enable bit for ecc_uncorrected_err_intr. ■ 1: Enabled ■ 0: Disabled **Value After Reset:** 0x1 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic
8	ecc_corrected_err_intr_en	R/W	Interrupt enable bit for ecc_corrected_err_intr. ■ 1 Enabled ■ 0 Disabled **Value After Reset:** 0x1 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic
7:5			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
4	ecc_ap_err_intr_clr	R/W1C	Interrupt clear bit for ecc_ap_err. If this bit is set, the ECCAPSTAT.ecc_ap_err/ecc_ap_err_intr will be cleared. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 && MEMC_ECCAP==1 **Testable:** readOnly **Programming Mode:** Dynamic
3	ecc_uncorr_err_cnt_clr	R/W1C	Setting this register bit to 1 clears the currently stored uncorrected ECC error count. The ECCERRCNT.ecc_uncorr_err_cnt register is cleared by this operation. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Testable:** readOnly **Programming Mode:** Dynamic
2	ecc_corr_err_cnt_clr	R/W1C	Setting this register bit to 1 clears the currently stored corrected ECC error count. The ECCERRCNT.ecc_corr_err_cnt register is cleared by this operation. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Testable:** readOnly **Programming Mode:** Dynamic
1	ecc_uncorrected_err_clr	R/W1C	Setting this register bit to 1 clears the currently stored uncorrected ECC error. The following registers are cleared: ■ ECCSTAT.ecc_uncorrected_err ■ ECCSTAT.sbr_read_ecc_ue ■ ADVECCSTAT.sbr_read_advecc_ue ■ ADVECCSTAT.advecc_uncorr_err_kbd_stat ■ ADVECCSTAT.advecc_uncorrected_err ■ ECCUSYN0 ■ ECCUSYN1 ■ ECCUSYN2 ■ ECCUDATA0 ■ ECCUDATA1 ■ ECCSYMBOL.ecc_uncorr_sym_71_64 DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
0	ecc_corrected_err_clr	R/W1C	Setting this register bit to 1 clears the currently stored corrected ECC error. The following registers are cleared: ■ ECCSTAT.ecc_corrected_err ■ ECCSTAT.sbr_read_ecc_ce ■ ADVECCSTAT.sbr_read_advecc_ce ■ ADVECCSTAT.advecc_corr_err_kbd_stat ■ ADVECCSTAT.advecc_corrected_err ■ ADVECCSTAT.advecc_num_err_symbol ■ ADVECCSTAT.advecc_err_symbol_pos ■ ADVECCSTAT.advecc_err_symbol_bits ■ ECCCSYN0 ■ ECCCSYN1 ■ ECCCSYN2 ■ ECCBITMASK0 ■ ECCBITMASK1 ■ ECCBITMASK2 ■ ECCCDATA0 ■ ECCCDATA1 ■ ECCSYMBOL.ecc_corr_sym_71_64 DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Testable:** readOnly **Programming Mode:** Dynamic

1.2.51 ECCERRCNT

- **Description:** ECC Error Counter Register
- **Size:** 32 bits
- **Offset:** 0x10610
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH0.

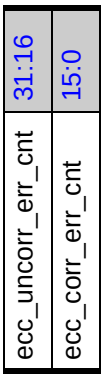


Table 1-110 Fields for Register: ECCERRCNT

Bits	Name	Memory Access	Description
31:16	ecc_uncorr_err_cnt	R	Number of uncorrectable ECC errors detected. It will saturates at 0xFFFF **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic
15:0	ecc_corr_err_cnt	R	Number of correctable ECC errors detected. It will saturates at 0xFFFF **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic

1.2.52 ECCCADDR0

- **Description:** ECC Corrected Error Address Register 0
- **Size:** 32 bits
- **Offset:** 0x10614
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH0.

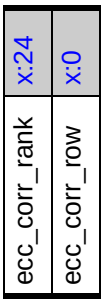


Table 1-111 Fields for Register: ECCCADDR0

Bits	Name	Memory Access	Description
x:24	ecc_corr_rank	R	Indicates the rank number of a read resulting in a corrected ECC error **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 && MEMC_NUM_RANKS>1 **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_RANK_BITS" + 23
x:0	ecc_corr_row	R	Indicates the page/row number of a read resulting in a corrected ECC error. This is 18-bits wide in configurations with LPDDR5 or DDR4 support and 16-bits in all other configurations. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_PAGE_BITS" - 1

1.2.53 ECCCADDR1

- **Description:** ECC Corrected Error Address Register 1
- **Size:** 32 bits
- **Offset:** 0x10618
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH0.

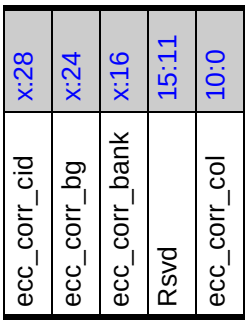


Table 1-112 Fields for Register: ECCCADDR1

Bits	Name	Memory Access	Description
x:28	ecc_corr_cid	R	CID number of a read resulting in a corrected ECC error. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 && UMCTL2_CID_EN==1 **Programming Mode:** Dynamic **Range Variable[x]:** "UMCTL2_CID_WIDTH" + 27
x:24	ecc_corr_bg	R	Bank Group number of a read resulting in a corrected ECC error. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_BG_BITS" + 23
x:16	ecc_corr_bank	R	Bank number of a read resulting in a corrected ECC error. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_BANK_BITS" + 15
15:11			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
10:0	ecc_corr_col	R	Block number of a read resulting in a corrected ECC error (lowest bit not assigned here). **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic

1.2.54 ECCCSYN0

- **Description:** ECC Corrected Syndrome Register 0
- **Size:** 32 bits
- **Offset:** 0x1061c
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH0.

Table 1-113 Fields for Register: ECCCSYN0

Bits	Name	Memory Access	Description
31:0	ecc_corr_syndromes_31_0	R	Data pattern that resulted in a corrected error. For 16-bit ECC, only bits [15:0] are used. This field can be masked by setting the dis_regs_ecc_syndrome input to value 1. This field will be cleared when ecc_corrected_err_clr is set. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic

1.2.55 ECCCSYN1

- **Description:** ECC Corrected Syndrome Register 1
- **Size:** 32 bits
- **Offset:** 0x10620
- **Exists:** MEMC_ECC_SUPPORT>0 && (MEMC_DRAM_DATA_WIDTH_64_OR_MEMC_INLINE_ECC==1)

This register is in block REGB_DDRC_CH0.

Table 1-114 Fields for Register: ECCCSYN1

Bits	Name	Memory Access	Description
31:0	ecc_corr_syndromes_63_32	R	Data pattern that resulted in a corrected error. For 32-bit ECC and 16-bit ECC, this register is not used. However, for multi-beat ECC, it represents the data pattern of odd SDRAM data beat (ECC lane). This field can be masked by setting the dis_regs_ecc_syndrome input to value 1.. This field will be cleared when ecc_corrected_err_clr is set. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 && (MEMC_DRAM_DATA_WIDTH_64_OR_MEMC_INLINE_ECC==1) **Programming Mode:** Dynamic

1.2.56 ECCCSYN2

- **Description:** ECC Corrected Syndrome Register 2
- **Size:** 32 bits
- **Offset:** 0x10624
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH0.

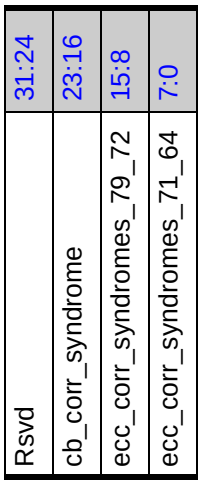


Table 1-115 Fields for Register: ECCCSYN2

Bits	Name	Memory Access	Description
31:24			**Reserved Field:** Yes
23:16	cb_corr_syndrome	R	Indicates the Checkbit corrected error syndrome that resulted in ECC error. It is computed by XOR operation of incoming checkbits and computed checkbits of the incoming data bits. It indicates which bit is in error, or whether multiple bits are in Error. This field will be cleared when ecc_corrected_err_clr is set. **Value After Reset:** 0x0 **Exists:** MEMC_SECDED_SIDEBAND_ECC==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
15:8	ecc_corr_syndromes_79_72	R	Indicates the data pattern that resulted in a corrected error. This register refers to the higher ECC byte, which is bits [79:72] for 64-bit ECC, [47:40] for 32-bit ECC, or [31:24] for 16-bit ECC. This field can be masked by setting the dis_regs_ecc_syndrome input to value 1. This field will be cleared when ecc_corrected_err_clr is set. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT==4 **Programming Mode:** Dynamic
7:0	ecc_corr_syndromes_71_64	R	Indicates the data pattern that resulted in a corrected error. This register refers to the ECC byte, which is bits [71:64] for 64-bit ECC, [39:32] for 32-bit ECC, or [23:16] for 16-bit ECC. This field can be masked by setting the dis_regs_ecc_syndrome input to value 1. This field will be cleared when ecc_corrected_err_clr is set. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic

1.2.57 ECCBITMASK0

- **Description:** ECC Corrected Data Bit Mask Register 0
- **Size:** 32 bits
- **Offset:** 0x10628
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH0.

Table 1-116 Fields for Register: ECCBITMASK0

Bits	Name	Memory Access	Description
31:0	ecc_corr_bit_mask_31_0	R	Mask for the corrected data portion ■ 1 on any bit indicates that the bit has been corrected by the ECC logic ■ 0 on any bit indicates that the bit has not been corrected by the ECC logic This register accumulates data over multiple ECC errors, to give an overall indication of which bits are being fixed. It is cleared by writing a 1 to ECCCTL.ecc_corrected_err_clr. For 16-bit ECC, only bits [15:0] are used **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic

1.2.58 ECCBITMASK1

- **Description:** ECC Corrected Data Bit Mask Register 1
- **Size:** 32 bits
- **Offset:** 0x1062c
- **Exists:** MEMC_ECC_SUPPORT>0 && (MEMC_DRAM_DATA_WIDTH_64_OR_MEMC_INLINE_ECC==1)

This register is in block REGB_DDRC_CH0.

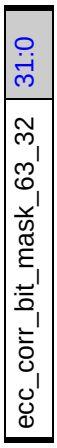


Table 1-117 Fields for Register: ECCBITMASK1

Bits	Name	Memory Access	Description
31:0	ecc_corr_bit_mask_63_32	R	Mask for the corrected data portion ■ 1 on any bit indicates that the bit has been corrected by the ECC logic ■ 0 on any bit indicates that the bit has not been corrected by the ECC logic This register accumulates data over multiple ECC errors, to give an overall indication of which bits are being fixed. It is cleared by writing a 1 to ECCCTL.ecc_corrected_err_clr. For 32-bit ECC and 16-bit ECC, this register is not used. However, for multi-beat ECC, it represents the ECC errors of odd SDRAM data beat (ECC lane). **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 && (MEMC_DRAM_DATA_WIDTH_64_OR_MEMC_INLINE_ECC==1) **Programming Mode:** Dynamic

1.2.59 ECCBITMASK2

- **Description:** ECC Corrected Data Bit Mask Register 2
- **Size:** 32 bits
- **Offset:** 0x10630
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH0.

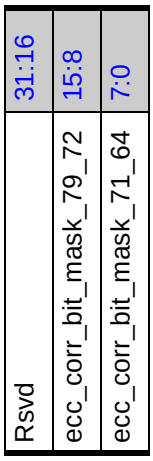


Table 1-118 Fields for Register: ECCBITMASK2

Bits	Name	Memory Access	Description
31:16			**Reserved Field:** Yes
15:8	ecc_corr_bit_mask_79_72	R	Indicates the mask for the corrected data portion. ■ 1 on any bit indicates that the bit has been corrected by the ECC logic ■ 0 on any bit indicates that the bit has not been corrected by the ECC logic This register accumulates data over multiple ECC errors, to give an overall indication of which bits are being fixed. It is cleared by writing a 1 to ECCCTL.ecc_corrected_err_clr. This register refers to the higher ECC byte, which is bits [79:72] for 64-bit ECC, [47:40] for 32-bit ECC, or [31:24] for 16-bit ECC. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT==4 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
7:0	ecc_corr_bit_mask_71_64	R	Mask for the corrected data portion ■ 1 on any bit indicates that the bit has been corrected by the ECC logic ■ 0 on any bit indicates that the bit has not been corrected by the ECC logic This register accumulates data over multiple ECC errors, to give an overall indication of which bits are being fixed. It is cleared by writing a 1 to ECCCTL.ecc_corrected_err_clr. This register refers to the ECC byte, which is bits [71:64] for 64-bit ECC, [39:32] for 32-bit ECC, or [23:16] for 16-bit ECC **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic

1.2.60 ECCUADDR0

- **Description:** ECC Uncorrected Error Address Register 0
- **Size:** 32 bits
- **Offset:** 0x10634
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH0.

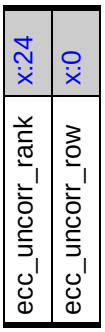


Table 1-119 Fields for Register: ECCUADDR0

Bits	Name	Memory Access	Description
x:24	ecc_uncorr_rank	R	Rank number of a read resulting in an uncorrected ECC error. **Value After Reset:** 0x0 **Exists:** MEMC_NUM_RANKS>1 **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_RANK_BITS" + 23
x:0	ecc_uncorr_row	R	Page/row number of a read resulting in an uncorrected ECC error. This is 18-bits wide in configurations with LPDDR5 or DDR4 support and 16-bits in all other configurations. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_PAGE_BITS" - 1

1.2.61 ECCUADDR1

- **Description:** ECC Uncorrected Error Address Register 1
- **Size:** 32 bits
- **Offset:** 0x10638
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH0.

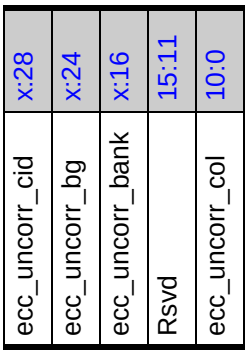


Table 1-120 Fields for Register: ECCUADDR1

Bits	Name	Memory Access	Description
x:28	ecc_uncorr_cid	R	CID number of a read resulting in an uncorrected ECC error. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 && UMCTL2_CID_EN==1 **Programming Mode:** Dynamic **Range Variable[x]:** "UMCTL2_CID_WIDTH" + 27
x:24	ecc_uncorr_bg	R	Bank Group number of a read resulting in an uncorrected ECC error **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_BG_BITS" + 23
x:16	ecc_uncorr_bank	R	Bank number of a read resulting in an uncorrected ECC error **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_BANK_BITS" + 15
15:11			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
10:0	ecc_uncorr_col	R	Block number of a read resulting in an uncorrected ECC error (lowest bit not assigned here) **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic

1.2.62 ECCUSYN0

- **Description:** ECC Uncorrected Syndrome Register 0
- **Size:** 32 bits
- **Offset:** 0x1063c
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH0.

Table 1-121 Fields for Register: ECCUSYN0

Bits	Name	Memory Access	Description
31:0	ecc_uncorr_syndromes_31_0	R	Data pattern that resulted in an uncorrected error. For 16-bit ECC, only bits [15:0] are used. This field can be masked by setting the dis_regs_ecc_syndrome input to value 1. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic

1.2.63 ECCUSYN1

- **Description:** ECC Uncorrected Syndrome Register 1
- **Size:** 32 bits
- **Offset:** 0x10640
- **Exists:** MEMC_ECC_SUPPORT>0 && (MEMC_DRAM_DATA_WIDTH_64_OR_MEMC_INLINE_ECC==1)

This register is in block REGB_DDRC_CH0.

Table 1-122 Fields for Register: ECCUSYN1

Bits	Name	Memory Access	Description
31:0	ecc_uncorr_syndromes_63_32	R	Data pattern that resulted in an uncorrected error. For 32-bit ECC and 16-bit ECC, this register is not used. However, for multi-beat ECC, it represents the data pattern of odd SDRAM data beat (ECC lane). This field can be masked by setting the dis_regs_ecc_syndrome input to value 1. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 && (MEMC_DRAM_DATA_WIDTH_64_OR_MEMC_INLINE_ECC==1) **Programming Mode:** Dynamic

1.2.64 ECCUSYN2

- **Description:** ECC Uncorrected Syndrome Register 2
- **Size:** 32 bits
- **Offset:** 0x10644
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH0.

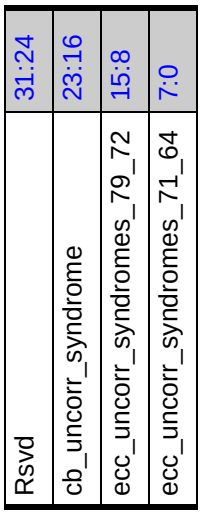


Table 1-123 Fields for Register: ECCUSYN2

Bits	Name	Memory Access	Description
31:24			**Reserved Field:** Yes
23:16	cb_uncorr_syndrome	R	Indicates the Checkbit uncorrected error syndrome that resulted in ECC error. . It is computed by XOR operation of incoming checkbits and computed checkbits of the incoming data bits. It indicates which bit is in error, or whether multiple bits are in Error. **Value After Reset:** 0x0 **Exists:** MEMC_SECDED_SIDEBAND_ECC==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
15:8	ecc_uncorr_syndromes_79_72	R	Data pattern that resulted in an uncorrected error. This register refers to the higher ECC byte, which is bits [79:72] for 64-bit ECC, [47:40] for 32-bit ECC, or [31:24] for 16-bit ECC. This field can be masked by setting the dis_regs_ecc_syndrome input to value 1. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT==4 **Programming Mode:** Dynamic
7:0	ecc_uncorr_syndromes_71_64	R	Data pattern that resulted in an uncorrected error. This register refers to the ECC byte, which is bits [71:64] for 64-bit ECC, [39:32] for 32-bit ECC, or [23:16] for 16-bit ECC. This field can be masked by setting the dis_regs_ecc_syndrome input to value 1. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic

1.2.65 ECCPOISONADDR0

- **Description:** ECC Data Poisoning Address Register 0.
- **Size:** 32 bits
- **Offset:** 0x10648
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH0.
If a HIF write data beat matches the address specified in this register, an ECC error will be introduced on that transaction (write/RMW), if ECCCFG1.data_poison_en=1

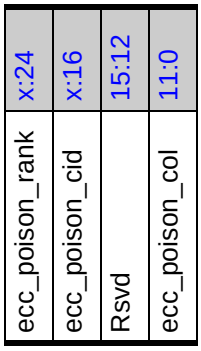


Table 1-124 Fields for Register: ECCPOISONADDR0

Bits	Name	Memory Access	Description
x:24	ecc_poison_rank	R/W	Rank address for ECC poisoning **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 && MEMC_NUM_RANKS>1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "MEMC_RANK_BITS" + 23
x:16	ecc_poison_cid	R/W	Indicates the chip ID for ECC poisoning (DDR4 3DS only) This register must be set to 0 when DDR4 3DS feature is not used. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 && UMCTL2_CID_EN==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "UMCTL2_CID_WIDTH" + 15
15:12			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
11:0	ecc_poison_col	R/W	Indicates the column address for ECC poisoning. Note that this column address must be burst aligned: ■ In full bus width mode, ecc_poison_col[2:0] must be set to 0 ■ In half bus width mode, ecc_poison_col[3:0] must be set to 0 ■ In quarter bus width mode, ecc_poison_col[4:0] must be set to 0 **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3

1.2.66 ECCPOISONADDR1

- **Description:** ECC Data Poisoning Address Register 1.
- **Size:** 32 bits
- **Offset:** 0x1064c
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH0.
If a HIF write data beat matches the address specified in this register, an ECC error will be introduced on that transaction (write/RMW), if ECCCFG1.data_poison_en=1

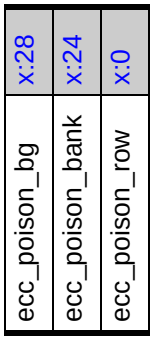


Table 1-125 Fields for Register: ECCPOISONADDR1

Bits	Name	Memory Access	Description
x:28	ecc_poison_bg	R/W	Bank Group address for ECC poisoning **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "MEMC_BG_BITS" + 27
x:24	ecc_poison_bank	R/W	Bank address for ECC poisoning **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "MEMC_BANK_BITS" + 23
x:0	ecc_poison_row	R/W	Row address for ECC poisoning. This is 18-bits wide in configurations with LPDDR5 or DDR4 support and 16-bits in all other configurations. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "MEMC_PAGE_BITS" - 1

1.2.67 ECCAPSTAT

- **Description:** Address protection within ECC Status Register
- **Size:** 32 bits
- **Offset:** 0x10664
- **Exists:** MEMC_ECCAP==1

This register is in block REGB_DDRC_CH0.

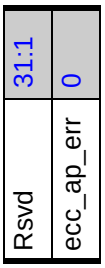


Table 1-126 Fields for Register: ECCAPSTAT

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	ecc_ap_err	R	Indicates the number of ECC errors (correctable/uncorrectable) within one burst exceeded the threshold.(ECCCFG0.ecc_ap_err_threshold) **Value After Reset:** 0x0 **Exists:** MEMC_ECCAP==1 **Programming Mode:** Static

1.2.68 OCPARCFG0

- **Description:** On-Chip Parity Configuration Register 0
- **Size:** 32 bits
- **Offset:** 0x10680
- **Exists:** UMCTL2_OCPAR_OR_OCECC_EN_1==1

This register is in block REGB_DDRC_CH0.

31:27	26	25	24	23	22	21	20	19:16	15	14	13	12	11:9	8	7	6	5	4	3:2	1	0
Rsvd	par_raddr_err_intr_force	par_waddr_err_intr_force	par_raddr_err_intr_clr	par_raddr_err_intr_en	par_waddr_err_intr_clr	par_waddr_err_intr_en	par_addr_slverr_en	Rsvd	par_rdata_err_intr_force	par_rdata_err_intr_clr	par_rdata_err_intr_en	par_rdata_slverr_en	Rsvd	par_wdata_axi_check_bypass_en	par_wdata_err_intr_force	par_wdata_err_intr_clr	par_wdata_slverr_en	par_wdata_err_intr_en	Rsvd	oc_parity_type	oc_parity_en

Table 1-127 Fields for Register: OCPARCFG0

Bits	Name	Memory Access	Description
31:27			**Reserved Field:** Yes
26	par_raddr_err_intr_force	R/W1C	Interrupt force bit for all par_raddr_err_intr. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
25	par_waddr_err_intr_force	R/W1C	Interrupt force bit for all par_waddr_err_intr. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
24	par_raddr_err_intr_clr	R/W1C	Interrupt clear bit for all par_raddr_err_intr. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
23	par_raddr_err_intr_en	R/W	Enables interrupt generation, if set to 1, for all ports, on signal par_raddr_err_intr_n upon detection of parity error on the AXI interface. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Dynamic
22	par_waddr_err_intr_clr	R/W1C	Interrupt clear bit for all par_waddr_err_intr. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
21	par_waddr_err_intr_en	R/W	Enables interrupt generation, if set to 1, for all ports, on signal par_waddr_err_intr_n upon detection of parity error on the AXI interface. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Dynamic
20	par_addr_slverr_en	R/W	Enables SLVERR generation on read response or write response when address parity error is detected at the AXI interface. **Value After Reset:** 0x1 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
19:16			**Reserved Field:** Yes
15	par_rdata_err_intr_force	R/W1C	Interrupt force bit for all par_rdata_err_intr_n and par_rdata_in_err_ecc_intr (Inline-ECC only). DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** UMCTL2_OCPAR_EN_1==1 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
14	par_rdata_err_intr_clr	R/W1C	Interrupt clear bit for par_rdata_err_intr. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** UMCTL2_OCPAR_EN_1==1 **Testable:** readOnly **Programming Mode:** Dynamic
13	par_rdata_err_intr_en	R/W	Enables interrupt generation, if set to 1, for all ports, on signal par_rdata_err_intr_n upon detection of parity error at the AXI interface. **Value After Reset:** 0x1 **Exists:** UMCTL2_OCPAR_EN_1==1 **Programming Mode:** Dynamic
12	par_rdata_slverr_en	R/W	Enables SLVERR generation on read response when read data parity error is detected at the AXI interface. **Value After Reset:** 0x1 **Exists:** UMCTL2_OCPAR_EN_1==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
11:9			**Reserved Field:** Yes
8	par_wdata_axi_check_bypass_en	R/W	Register to bypass write data parity checker at AXI. If set to 1, write data parity checker at AXI is bypassed - incoming write data parity error or poisoned write data parity will be propagated to next block of the design. If set to 0, write data parity checker at AXI will detect the incoming write data parity error and report it onto the interrupt and bad parity is terminated - correct parity is propagated. **Value After Reset:** 0x0 **Exists:** UMCTL2_OCPAR_EN_1==1 **Programming Mode:** Quasi-dynamic Group 3
7	par_wdata_err_intr_force	R/W1C	Interrupt force bit for par_wdata_err_intr. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** UMCTL2_OCPAR_EN_1==1 **Testable:** readOnly **Programming Mode:** Dynamic
6	par_wdata_err_intr_clr	R/W1C	Interrupt clear bit for par_wdata_err_intr. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** UMCTL2_OCPAR_EN_1==1 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
5	par_wdata_slverr_en	R/W	Enables SLVERR generation on write response when write data parity error is detected at the AXI interface. **Value After Reset:** 0x1 **Exists:** UMCTL2_OCPAR_EN_1==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
4	par_wdata_err_intr_en	R/W	Enables write data interrupt generation (par_wdata_err_intr) upon detection of parity error at the AXI or DFI interface. **Value After Reset:** 0x1 **Exists:** UMCTL2_OCPAR_EN_1==1 **Volatile:** true **Programming Mode:** Dynamic
3:2			**Reserved Field:** Yes
1	oc_parity_type	R/W	Parity type: ■ 0: Even parity ■ 1: Odd parity **Value After Reset:** 0x1 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
0	oc_parity_en	R/W	Parity enable register. Enables On-Chip parity for all interfaces. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3

1.2.69 OCPARCFG1

- **Description:** On-Chip Parity Configuration Register 1
- **Size:** 32 bits
- **Offset:** 0x10684
- **Exists:** UMCTL2_OCPAR_EN_1==1

This register is in block REGB_DDRC_CH0.

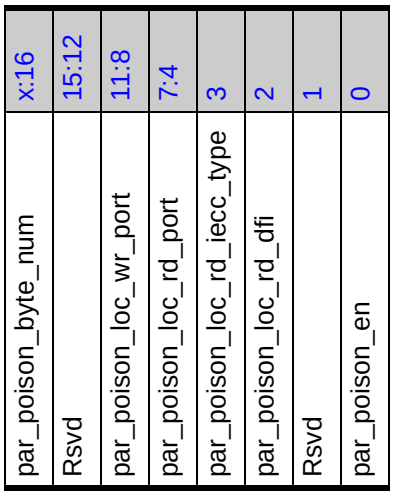


Table 1-128 Fields for Register: OCPARCFG1

Bits	Name	Memory Access	Description
x:16	par_poison_byte_num	R/W	Indicates the byte number (binary encoded) where the parity error is to be injected at the read data AXI interface. Error can be injected one byte at a time. **Value After Reset:** 0x0 **Exists:** MEMC_INLINE_ECC==0 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "UMCTL2_DATARAM_PAR_DW_LG2" + 15
15:12			**Reserved Field:** Yes
11:8	par_poison_loc_wr_port	R/W	Enables parity poisoning on write data at the AXI interface before the input parity check logic. The value specifies the binary encoded port number of the AXI interface to be injected with parity error. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3

Bits	Name	Memory Access	Description
7:4	par_poison_loc_rd_port	R/W	Enables parity poisoning on read data at the AXI interface after the parity check logic. The value specifies the binary encoded port number of the AXI interface to be injected with parity error. Error can be injected to one port at a time. An error injected here is not logged and does not trigger SLVERR or interrupt by the controller. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
3	par_poison_loc_rd_iecc_type	R/W	Selects which parity to poison at the DFI when inline ECC is enabled. If this register is set to 0, parity error is injected on the first read data going through the ECC path; if this register is set to 1, parity error is injected on the first read data going through the data path. **Value After Reset:** 0x0 **Exists:** MEMC_INLINE_ECC==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
2	par_poison_loc_rd_dfi	R/W	Enables parity poisoning on read data at the DFI interface after the parity generation logic, and when MEMC_INLINE_ECC=1 enables poisoning of ECC word after the ECC encoder at the write data interface at the DFI. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
1			**Reserved Field:** Yes
0	par_poison_en	R/W	Enables on-chip parity poisoning on the data interfaces. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3

1.2.70 OCPARSTAT0

- **Description:** On-Chip Parity Status Register 0
- **Size:** 32 bits
- **Offset:** 0x10688
- **Exists:** UMCTL2_OCPAR_OR_OCECC_EN_1==1

This register is in block REGB_DDRC_CH0.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
par_raddr_err_intr_15	par_raddr_err_intr_14	par_raddr_err_intr_13	par_raddr_err_intr_12	par_raddr_err_intr_11	par_raddr_err_intr_10	par_raddr_err_intr_9	par_raddr_err_intr_8	par_raddr_err_intr_7	par_raddr_err_intr_6	par_raddr_err_intr_5	par_raddr_err_intr_4	par_raddr_err_intr_3	par_raddr_err_intr_2	par_raddr_err_intr_1	par_raddr_err_intr_0	par_waddr_err_intr_15	par_waddr_err_intr_14	par_waddr_err_intr_13	par_waddr_err_intr_12	par_waddr_err_intr_11	par_waddr_err_intr_10	par_waddr_err_intr_9	par_waddr_err_intr_8	par_waddr_err_intr_7	par_waddr_err_intr_6	par_waddr_err_intr_5	par_waddr_err_intr_4	par_waddr_err_intr_3	par_waddr_err_intr_2	par_waddr_err_intr_1	par_waddr_err_intr_0

Table 1-129 Fields for Register: OCPARSTAT0

Bits	Name	Memory Access	Description
31	par_raddr_err_intr_15	R	Read address parity error interrupt for port 15. This interrupt is asserted when an on-chip read address parity error occurred on the corresponding AXI port's read address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_15==1 **Programming Mode:** Static
30	par_raddr_err_intr_14	R	Read address parity error interrupt for port 14. This interrupt is asserted when an on-chip read address parity error occurred on the corresponding AXI port's read address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_14==1 **Programming Mode:** Static
29	par_raddr_err_intr_13	R	Read address parity error interrupt for port 13. This interrupt is asserted when an on-chip read address parity error occurred on the corresponding AXI port's read address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_13==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
28	par_raddr_err_intr_12	R	Read address parity error interrupt for port 12. This interrupt is asserted when an on-chip read address parity error occurred on the corresponding AXI port's read address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_12==1 **Programming Mode:** Static
27	par_raddr_err_intr_11	R	Read address parity error interrupt for port 11. This interrupt is asserted when an on-chip read address parity error occurred on the corresponding AXI port's read address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_11==1 **Programming Mode:** Static
26	par_raddr_err_intr_10	R	Read address parity error interrupt for port 10. This interrupt is asserted when an on-chip read address parity error occurred on the corresponding AXI port's read address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_10==1 **Programming Mode:** Static
25	par_raddr_err_intr_9	R	Read address parity error interrupt for port 9. This interrupt is asserted when an on-chip read address parity error occurred on the corresponding AXI port's read address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_9==1 **Programming Mode:** Static
24	par_raddr_err_intr_8	R	Read address parity error interrupt for port 8. This interrupt is asserted when an on-chip read address parity error occurred on the corresponding AXI port's read address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_8==1 **Programming Mode:** Static
23	par_raddr_err_intr_7	R	Read address parity error interrupt for port 7. This interrupt is asserted when an on-chip read address parity error occurred on the corresponding AXI port's read address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_7==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
22	par_raddr_err_intr_6	R	Read address parity error interrupt for port 6. This interrupt is asserted when an on-chip read address parity error occurred on the corresponding AXI port's read address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_6==1 **Programming Mode:** Static
21	par_raddr_err_intr_5	R	Read address parity error interrupt for port 5. This interrupt is asserted when an on-chip read address parity error occurred on the corresponding AXI port's read address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_5==1 **Programming Mode:** Static
20	par_raddr_err_intr_4	R	Read address parity error interrupt for port 4. This interrupt is asserted when an on-chip read address parity error occurred on the corresponding AXI port's read address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_4==1 **Programming Mode:** Static
19	par_raddr_err_intr_3	R	Read address parity error interrupt for port 3. This interrupt is asserted when an on-chip read address parity error occurred on the corresponding AXI port's read address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_3==1 **Programming Mode:** Static
18	par_raddr_err_intr_2	R	Read address parity error interrupt for port 2. This interrupt is asserted when an on-chip read address parity error occurred on the corresponding AXI port's read address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_2==1 **Programming Mode:** Static
17	par_raddr_err_intr_1	R	Read address parity error interrupt for port 1. This interrupt is asserted when an on-chip read address parity error occurred on the corresponding AXI port's read address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_1==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
16	par_raddr_err_intr_0	R	Read address parity error interrupt for port 0. This interrupt is asserted when an on-chip read address parity error occurred on the corresponding AXI port's read address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_0==1 **Programming Mode:** Static
15	par_waddr_err_intr_15	R	Write address parity error interrupt for port 15. This interrupt is asserted when an on-chip write address parity error occurred on the corresponding AXI port's write address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_15==1 **Programming Mode:** Static
14	par_waddr_err_intr_14	R	Write address parity error interrupt for port 14. This interrupt is asserted when an on-chip write address parity error occurred on the corresponding AXI port's write address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_14==1 **Programming Mode:** Static
13	par_waddr_err_intr_13	R	Write address parity error interrupt for port 13. This interrupt is asserted when an on-chip write address parity error occurred on the corresponding AXI port's write address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_13==1 **Programming Mode:** Static
12	par_waddr_err_intr_12	R	Write address parity error interrupt for port 12. This interrupt is asserted when an on-chip write address parity error occurred on the corresponding AXI port's write address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_12==1 **Programming Mode:** Static
11	par_waddr_err_intr_11	R	Write address parity error interrupt for port 11. This interrupt is asserted when an on-chip write address parity error occurred on the corresponding AXI port's write address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_11==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
10	par_waddr_err_intr_10	R	Write address parity error interrupt for port 10. This interrupt is asserted when an on-chip write address parity error occurred on the corresponding AXI port's write address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_10==1 **Programming Mode:** Static
9	par_waddr_err_intr_9	R	Write address parity error interrupt for port 9. This interrupt is asserted when an on-chip write address parity error occurred on the corresponding AXI port's write address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_9==1 **Programming Mode:** Static
8	par_waddr_err_intr_8	R	Write address parity error interrupt for port 8. This interrupt is asserted when an on-chip write address parity error occurred on the corresponding AXI port's write address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_8==1 **Programming Mode:** Static
7	par_waddr_err_intr_7	R	Write address parity error interrupt for port 7. This interrupt is asserted when an on-chip write address parity error occurred on the corresponding AXI port's write address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_7==1 **Programming Mode:** Static
6	par_waddr_err_intr_6	R	Write address parity error interrupt for port 6. This interrupt is asserted when an on-chip write address parity error occurred on the corresponding AXI port's write address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_6==1 **Programming Mode:** Static
5	par_waddr_err_intr_5	R	Write address parity error interrupt for port 5. This interrupt is asserted when an on-chip write address parity error occurred on the corresponding AXI port's write address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_5==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
4	par_waddr_err_intr_4	R	Write address parity error interrupt for port 4. This interrupt is asserted when an on-chip write address parity error occurred on the corresponding AXI port's write address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_4==1 **Programming Mode:** Static
3	par_waddr_err_intr_3	R	Write address parity error interrupt for port 3. This interrupt is asserted when an on-chip write address parity error occurred on the corresponding AXI port's write address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_3==1 **Programming Mode:** Static
2	par_waddr_err_intr_2	R	Write address parity error interrupt for port 2. This interrupt is asserted when an on-chip write address parity error occurred on the corresponding AXI port's write address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_2==1 **Programming Mode:** Static
1	par_waddr_err_intr_1	R	Write address parity error interrupt for port 1. This interrupt is asserted when an on-chip write address parity error occurred on the corresponding AXI port's write address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_1==1 **Programming Mode:** Static
0	par_waddr_err_intr_0	R	Write address parity error interrupt for port 0. This interrupt is asserted when an on-chip write address parity error occurred on the corresponding AXI port's write address channel. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_0==1 **Programming Mode:** Static

1.2.71 OCPARSTAT1

- **Description:** On-Chip Parity Status Register 1
- **Size:** 32 bits
- **Offset:** 0x1068c
- **Exists:** UMCTL2_OCPAR_EN_1==1

This register is in block REGB_DDRC_CH0.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
par_rdata_err_intr_15	par_rdata_err_intr_14	par_rdata_err_intr_13	par_rdata_err_intr_12	par_rdata_err_intr_11	par_rdata_err_intr_10	par_rdata_err_intr_9	par_rdata_err_intr_8	par_rdata_err_intr_7	par_rdata_err_intr_6	par_rdata_err_intr_5	par_rdata_err_intr_4	par_rdata_err_intr_3	par_rdata_err_intr_2	par_rdata_err_intr_1	par_rdata_err_intr_0	par_wdata_in_err_intr_15	par_wdata_in_err_intr_14	par_wdata_in_err_intr_13	par_wdata_in_err_intr_12	par_wdata_in_err_intr_11	par_wdata_in_err_intr_10	par_wdata_in_err_intr_9	par_wdata_in_err_intr_8	par_wdata_in_err_intr_7	par_wdata_in_err_intr_6	par_wdata_in_err_intr_5	par_wdata_in_err_intr_4	par_wdata_in_err_intr_3	par_wdata_in_err_intr_2	par_wdata_in_err_intr_1	par_wdata_in_err_intr_0

Table 1-130 Fields for Register: OCPARSTAT1

Bits	Name	Memory Access	Description
31	par_rdata_err_intr_15	R	Read data parity error interrupt for port 15. This interrupt is asserted when an on-chip read data parity error occurred on the corresponding AXI port's read data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_rdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_15==1 **Programming Mode:** Static
30	par_rdata_err_intr_14	R	Read data parity error interrupt for port 14. This interrupt is asserted when an on-chip read data parity error occurred on the corresponding AXI port's read data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_rdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_14==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
29	par_rdata_err_intr_13	R	Read data parity error interrupt for port 13. This interrupt is asserted when an on-chip read data parity error occurred on the corresponding AXI port's read data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_rdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_13==1 **Programming Mode:** Static
28	par_rdata_err_intr_12	R	Read data parity error interrupt for port 12. This interrupt is asserted when an on-chip read data parity error occurred on the corresponding AXI port's read data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_rdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_12==1 **Programming Mode:** Static
27	par_rdata_err_intr_11	R	Read data parity error interrupt for port 11. This interrupt is asserted when an on-chip read data parity error occurred on the corresponding AXI port's read data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_rdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_11==1 **Programming Mode:** Static
26	par_rdata_err_intr_10	R	Read data parity error interrupt for port 10. This interrupt is asserted when an on-chip read data parity error occurred on the corresponding AXI port's read data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_rdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_10==1 **Programming Mode:** Static
25	par_rdata_err_intr_9	R	Read data parity error interrupt for port 9. This interrupt is asserted when an on-chip read data parity error occurred on the corresponding AXI port's read data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_rdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_9==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
24	par_rdata_err_intr_8	R	Read data parity error interrupt for port 8. This interrupt is asserted when an on-chip read data parity error occurred on the corresponding AXI port's read data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_rdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_8==1 **Programming Mode:** Static
23	par_rdata_err_intr_7	R	Read data parity error interrupt for port 7. This interrupt is asserted when an on-chip read data parity error occurred on the corresponding AXI port's read data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_rdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_7==1 **Programming Mode:** Static
22	par_rdata_err_intr_6	R	Read data parity error interrupt for port 6. This interrupt is asserted when an on-chip read data parity error occurred on the corresponding AXI port's read data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_rdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_6==1 **Programming Mode:** Static
21	par_rdata_err_intr_5	R	Read data parity error interrupt for port 5. This interrupt is asserted when an on-chip read data parity error occurred on the corresponding AXI port's read data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_rdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_5==1 **Programming Mode:** Static
20	par_rdata_err_intr_4	R	Read data parity error interrupt for port 4. This interrupt is asserted when an on-chip read data parity error occurred on the corresponding AXI port's read data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_rdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_4==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
19	par_rdata_err_intr_3	R	Read data parity error interrupt for port 3. This interrupt is asserted when an on-chip read data parity error occurred on the corresponding AXI port's read data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_rdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_3==1 **Programming Mode:** Static
18	par_rdata_err_intr_2	R	Read data parity error interrupt for port 2. This interrupt is asserted when an on-chip read data parity error occurred on the corresponding AXI port's read data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_rdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_2==1 **Programming Mode:** Static
17	par_rdata_err_intr_1	R	Read data parity error interrupt for port 1. This interrupt is asserted when an on-chip read data parity error occurred on the corresponding AXI port's read data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_rdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_1==1 **Programming Mode:** Static
16	par_rdata_err_intr_0	R	Read data parity error interrupt for port 0. This interrupt is asserted when an on-chip read data parity error occurred on the corresponding AXI port's read data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_rdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_0==1 **Programming Mode:** Static
15	par_wdata_in_err_intr_15	R	Write data parity error interrupt on input for port 15. This interrupt is asserted when an on-chip write data parity error occurred on the corresponding AXI port's write data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_wdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_15==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
14	par_wdata_in_err_intr_14	R	Write data parity error interrupt on input for port 14. This interrupt is asserted when an on-chip write data parity error occurred on the corresponding AXI port's write data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_wdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_14==1 **Programming Mode:** Static
13	par_wdata_in_err_intr_13	R	Write data parity error interrupt on input for port 13. This interrupt is asserted when an on-chip write data parity error occurred on the corresponding AXI port's write data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_wdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_13==1 **Programming Mode:** Static
12	par_wdata_in_err_intr_12	R	Write data parity error interrupt on input for port 12. This interrupt is asserted when an on-chip write data parity error occurred on the corresponding AXI port's write data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_wdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_12==1 **Programming Mode:** Static
11	par_wdata_in_err_intr_11	R	Write data parity error interrupt on input for port 11. This interrupt is asserted when an on-chip write data parity error occurred on the corresponding AXI port's write data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_wdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_11==1 **Programming Mode:** Static
10	par_wdata_in_err_intr_10	R	Write data parity error interrupt on input for port 10. This interrupt is asserted when an on-chip write data parity error occurred on the corresponding AXI port's write data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_wdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_10==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
9	par_wdata_in_err_intr_9	R	Write data parity error interrupt on input for port 9. This interrupt is asserted when an on-chip write data parity error occurred on the corresponding AXI port's write data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_wdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_9==1 **Programming Mode:** Static
8	par_wdata_in_err_intr_8	R	Write data parity error interrupt on input for port 8. This interrupt is asserted when an on-chip write data parity error occurred on the corresponding AXI port's write data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_wdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_8==1 **Programming Mode:** Static
7	par_wdata_in_err_intr_7	R	Write data parity error interrupt on input for port 7. This interrupt is asserted when an on-chip write data parity error occurred on the corresponding AXI port's write data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_wdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_7==1 **Programming Mode:** Static
6	par_wdata_in_err_intr_6	R	Write data parity error interrupt on input for port 6. This interrupt is asserted when an on-chip write data parity error occurred on the corresponding AXI port's write data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_wdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_6==1 **Programming Mode:** Static
5	par_wdata_in_err_intr_5	R	Write data parity error interrupt on input for port 5. This interrupt is asserted when an on-chip write data parity error occurred on the corresponding AXI port's write data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_wdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_5==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
4	par_wdata_in_err_intr_4	R	Write data parity error interrupt on input for port 4. This interrupt is asserted when an on-chip write data parity error occurred on the corresponding AXI port's write data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_wdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_4==1 **Programming Mode:** Static
3	par_wdata_in_err_intr_3	R	Write data parity error interrupt on input for port 3. This interrupt is asserted when an on-chip write data parity error occurred on the corresponding AXI port's write data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_wdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_3==1 **Programming Mode:** Static
2	par_wdata_in_err_intr_2	R	Write data parity error interrupt on input for port 2. This interrupt is asserted when an on-chip write data parity error occurred on the corresponding AXI port's write data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_wdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_2==1 **Programming Mode:** Static
1	par_wdata_in_err_intr_1	R	Write data parity error interrupt on input for port 1. This interrupt is asserted when an on-chip write data parity error occurred on the corresponding AXI port's write data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_wdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_1==1 **Programming Mode:** Static
0	par_wdata_in_err_intr_0	R	Write data parity error interrupt on input for port 0. This interrupt is asserted when an on-chip write data parity error occurred on the corresponding AXI port's write data channel. Bit 0 corresponds to Port 0, and so on. Cleared by register par_wdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_0==1 **Programming Mode:** Static

1.2.72 OCPARSTAT2

- **Description:** On-Chip Parity Status Register 2
- **Size:** 32 bits
- **Offset:** 0x10690
- **Exists:** UMCTL2_OCPAR_EN_1==1

This register is in block REGB_DDRC_CH0.

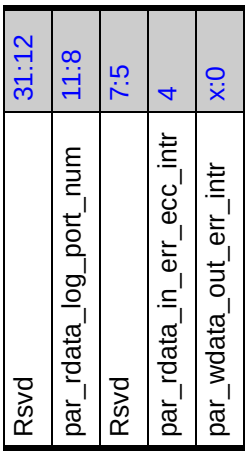


Table 1-131 Fields for Register: OCPARSTAT2

Bits	Name	Memory Access	Description
31:12			**Reserved Field:** Yes
11:8	par_rdata_log_port_num	R	Failing port number (binary encoded) of the last read data beat which resulted in on-chip parity error at the AXI interface. If there are more than one simultaneous port failures, the lower-indexed port is captured. **Value After Reset:** 0x0 **Exists:** MEMC_INLINE_ECC==0 **Programming Mode:** Static
7:5			**Reserved Field:** Yes
4	par_rdata_in_err_ecc_intr	R	Interrupt on ECC data going into inline ECC decoder. Cleared by par_rdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** MEMC_INLINE_ECC==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
x:0	par_wdata_out_err_intr	R	Write data parity error interrupt on output. Cleared by register par_wdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static **Range Variable[x]:** "UMCTL2_OCPAR_WDATA_OUT_ERR_WIDTH" - 1

1.2.73 OCPARSTAT3

- **Description:** On-Chip Parity Read Data Log Register 0
- **Size:** 32 bits
- **Offset:** 0x10694
- **Exists:** UMCTL2_OCPAR_EN_1==1 && MEMC_INLINE_ECC==0

This register is in block REGB_DDRC_CH0.

Table 1-132 Fields for Register: OCPARSTAT3

Bits	Name	Memory Access	Description
31:0	par_rdata_log_byte_num	R	Failing byte(s) number of the last read data beat which resulted in on-chip parity error at the AXI interface. This log reports failing bytes up to byte 31. In case, if the number of bytes at the AXI interface is less than 32, the unused upper par_rdata_log_byte_num bits are driven to '0' and must be ignored. For AXI interface with 64 bytes data width, the upper log byte number are reflected in OCPARSTAT8 register. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.2.74 OCPARSTAT4

- **Description:** On-Chip Parity Write Address Log Register 0
- **Size:** 32 bits
- **Offset:** 0x10698
- **Exists:** UMCTL2_OCPAR_EN_1==1 && MEMC_INLINE_ECC==0

This register is in block REGB_DDRC_CH0.

Table 1-133 Fields for Register: OCPARSTAT4

Bits	Name	Memory Access	Description
31:0	par_waddr_log_low	R	AXI system address [31:0] of the last write transaction resulting in on-chip parity error on write address path at the AXI interface. Depending on the crossing delay, for back-to-back errors, last address may not be logged, instead the first address logged will be kept. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.2.75 OCPARSTAT5

- **Description:** On-Chip Parity Write Address Log Register 1
- **Size:** 32 bits
- **Offset:** 0x1069c
- **Exists:** UMCTL2_OCPAR_EN_1==1 && MEMC_INLINE_ECC==0

This register is in block REGB_DDRC_CH0.

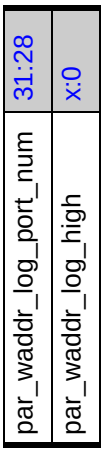


Table 1-134 Fields for Register: OCPARSTAT5

Bits	Name	Memory Access	Description
31:28	par_waddr_log_port_num	R	Failing port number (binary encoded) of the last write address transaction which resulted in on-chip parity error at the AXI interface. If there are more than one simultaneous port failures, the lower-indexed port is captured. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
x:0	par_waddr_log_high	R	AXI system address [59:32] of the last write transaction resulting in on-chip parity error on write address path at the AXI interface. **Value After Reset:** 0x0 **Exists:** UMCTL2_OCPAR_ADDR_LOG_USE_MSB==1 **Programming Mode:** Static **Range Variable[x]:** "UMCTL2_OCPAR_ADDR_LOG_HIGH_WIDTH" - 1

1.2.76 OCPARSTAT6

- **Description:** On-Chip Parity Read Address Log Register 0
- **Size:** 32 bits
- **Offset:** 0x106a0
- **Exists:** UMCTL2_OCPAR_EN_1==1 && MEMC_INLINE_ECC==0

This register is in block REGB_DDRC_CH0.

Table 1-135 Fields for Register: OCPARSTAT6

Bits	Name	Memory Access	Description
31:0	par_raddr_log_low	R	AXI system address [31:0] of the last read transaction resulting in on-chip parity error on read address path at the AXI interface. Depending on the crossing delay, for back-to-back errors, last address may not be logged, instead the first address logged will be kept. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.2.77 OCPARSTAT7

- **Description:** On-Chip Parity Read Address Log Register 1
- **Size:** 32 bits
- **Offset:** 0x106a4
- **Exists:** UMCTL2_OCPAR_EN_1==1 && MEMC_INLINE_ECC==0

This register is in block REGB_DDRC_CH0.

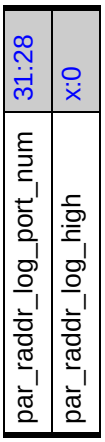


Table 1-136 Fields for Register: OCPARSTAT7

Bits	Name	Memory Access	Description
31:28	par_raddr_log_port_num	R	Failing port number (binary encoded) of the last read address transaction which resulted in on-chip parity error at the AXI interface. If there are more than one simultaneous port failures, the lower-indexed port is captured. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
x:0	par_raddr_log_high	R	AXI system address [59:32] of the last read transaction resulting in on-chip parity error on read address path at the AXI interface. **Value After Reset:** 0x0 **Exists:** UMCTL2_OCPAR_ADDR_LOG_USE_MSB==1 **Programming Mode:** Static **Range Variable[x]:** "UMCTL2_OCPAR_ADDR_LOG_HIGH_WIDTH" - 1

1.2.78 OCPARSTAT8

- **Description:** On-Chip Parity Read Data Log Register 1
- **Size:** 32 bits
- **Offset:** 0x106a8
- **Exists:** UMCTL2_OCPAR_EN_1==1 && MEMC_INLINE_ECC==0 && UMCTL2_MAX_X-PI_PORT_DW_GTEQ_512==1

This register is in block REGB_DDRC_CH0.

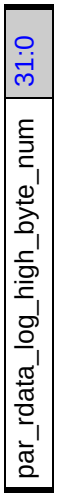


Table 1-137 Fields for Register: OCPARSTAT8

Bits	Name	Memory Access	Description
31:0	par_rdata_log_high_byte_num	R	Failing byte(s) number of the last read data beat which resulted in on-chip parity error at the AXI interface. This log reports failing bytes from byte 32 to byte 64. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.2.79 OCSAPCFG0

- **Description:** On-Chip external SRAM Address Protection Configuration Register 0
- **Size:** 32 bits
- **Offset:** 0x106b0
- **Exists:** DDRCTL_OCSAP_EN_1==1

This register is in block REGB_DDRC_CH0.

31:28	27	26:24	23:19	18:16	15:14	13	12	11:9	8	7:1	0
rdataram_addr_poison_port	Rsvd	rdataram_addr_poison_ctl	Rsvd	wdataram_addr_poison_ctl	Rsvd	rdataram_addr_poison_loc	wdataram_addr_poison_loc	Rsvd	ocsap_poison_en	Rsvd	ocsap_par_en

Table 1-138 Fields for Register: OCSAPCFG0

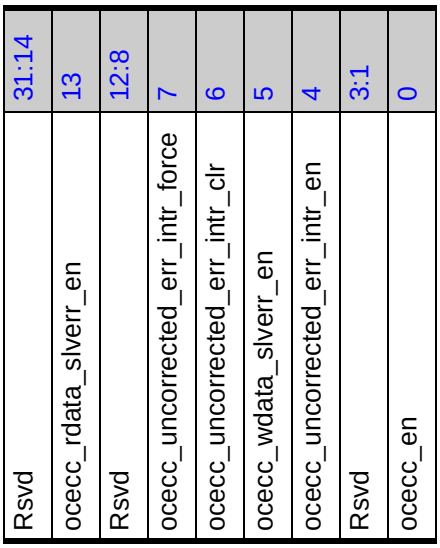
Bits	Name	Memory Access	Description
31:28	rdataram_addr_poison_port	R/W	The value specifies the binary encoded port number of the AXI interface for poisoning the associated RDATARAM address. Error can be injected to one port at a time. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
27			**Reserved Field:** Yes
26:24	rdataram_addr_poison_ctl	R/W	Selects rdataram address bit position (binary encoded) for poisoning. Note: The selected rdataram address bit position (binary encoded) shall be less than parameter value UMCTL2_RDATARAM_AW. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
23:19			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
18:16	wdataram_addr_poison_ctl	R/W	Selects wdataram address bit position (binary encoded) for poisoning. Note: The selected wdataram address bit position (binary encoded) shall be less than parameter value UMCTL2_WDATARAM_AW. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
15:14			**Reserved Field:** Yes
13	rdataram_addr_poison_loc	R/W	Selects rdataram address poisoning either for write address or read address. ■ 0 - Poison SRAM write address ■ 1 - Poison SRAM read address **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
12	wdataram_addr_poison_loc	R/W	Selects wdataram address poisoning either for write address or read address. ■ 0 - Poison SRAM write address ■ 1 - Poison SRAM read address **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
11:9			**Reserved Field:** Yes
8	ocsap_poison_en	R/W	Enables On-Chip external SRAM address poisoning. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
7:1			**Reserved Field:** Yes
0	ocsap_par_en	R/W	Enables On-Chip external SRAM address parity. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3

1.2.80 OCECCCFG0

- **Description:** On-Chip ECC Configuration Register 0
- **Size:** 32 bits
- **Offset:** 0x10700
- **Exists:** UMCTL2_OCECC_EN_1==1

This register is in block REGB_DDRC_CH0.



31:14	13	12:8	7	6	5	4	3:1	0
Rsvd	ococc_rdata_slverr_en	Rsvd	ococc_uncorrected_err_intr_force	ococc_uncorrected_err_intr_clr	ococc_wdata_slverr_en	ococc_uncorrected_err_intr_en	Rsvd	ococc_en

Table 1-139 Fields for Register: OCECCCFG0

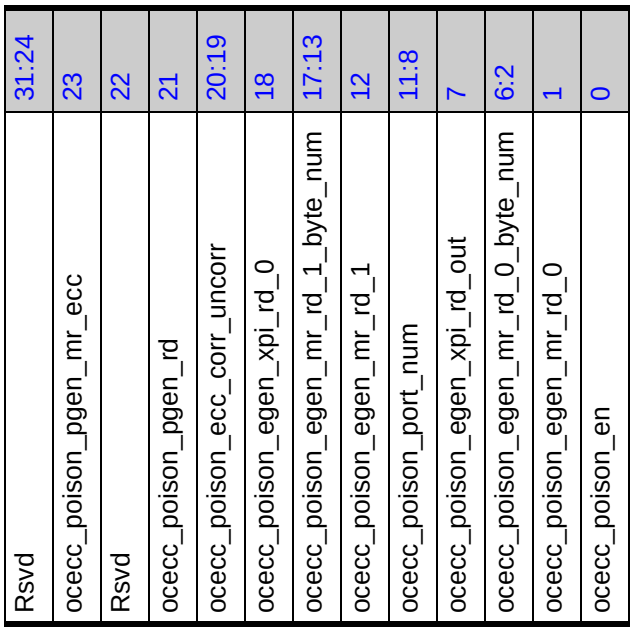
Bits	Name	Memory Access	Description
31:14			**Reserved Field:** Yes
13	ococc_rdata_slverr_en	R/W	Enables SLVERR generation on read responses. **Value After Reset:** 0x1 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
12:8			**Reserved Field:** Yes
7	ococc_uncorrected_err_intr_force	R/W1C	Interrupt force bit for ococc_uncorrected_err_intr. When this bit is set to 1, the OCECCSTAT0.ococc_uncorrected_err field and the ococc_uncorrected_err_intr pin will be set. The DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
6	ococc_uncorrected_err_intr_clr	R/W1C	Interrupt clear bit for ococc_uncorrected_err_intr. The DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
5	ococc_wdata_slverr_en	R/W	Enables SLVERR generation on write responses. **Value After Reset:** 0x1 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
4	ococc_uncorrected_err_intr_en	R/W	Enables uncorrected error interrupt generation. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Dynamic
3:1			**Reserved Field:** Yes
0	ococc_en	R/W	OCECC enable register. Enables On-Chip ECC for all interfaces. Note - OCPARCFG0.oc_parity_en register must be set to 1 to enable On-chip ECC functionality. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3

1.2.81 OCECCCFG1

- **Description:** On-Chip ECC Configuration Register 1
- **Size:** 32 bits
- **Offset:** 0x10704
- **Exists:** UMCTL2_OCECC_EN_1==1

This register is in block REGB_DDRC_CH0.



31:24	23	22	21	20:19	18	17:13	12	11:8	7	6:2	1	0
Rsvd	ocecc_poison_pgen_mr_ecc	Rsvd	ocecc_poison_pgen_rd	ocecc_poison_ecc_corr_uncorr	ocecc_poison_egen_xpi_rd_0	ocecc_poison_egen_mr_rd_1_byte_num	ocecc_poison_egen_mr_rd_1	ocecc_poison_port_num	ocecc_poison_egen_xpi_rd_out	ocecc_poison_egen_mr_rd_0_byte_num	ocecc_poison_egen_mr_rd_0	ocecc_poison_en

Table 1-140 Fields for Register: OCECCCFG1

Bits	Name	Memory Access	Description
31:24			**Reserved Field:** Yes
23	ocecc_poison_pgen_mr_ecc	R/W	Poisons parity for write ECC data. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
22			**Reserved Field:** Yes
21	ocecc_poison_pgen_rd	R/W	Poisons parity for read ECC data. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3

Bits	Name	Memory Access	Description
20:19	ococc_poison_ecc_corr_uncorr	R/W	Selects either to inject 1 or 2 bit error in one of the ECC encoders. ■ 1 Injects single bit error ■ 2 Injects double bit error ■ 0 and 3 unused **Value After Reset:** 0x2 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
18	ococc_poison_egen_xpi_rd_0	R/W	Poisons the ECC encoder for the read data of a Read command. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
17:13	ococc_poison_egen_mr_rd_1_byte_num	R/W	Byte number to poison for the read data of an RMW command ECC encoder. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
12	ococc_poison_egen_mr_rd_1	R/W	Poisons the ECC encoder for the read data of an RMW command. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
11:8	ococc_poison_port_num	R/W	Selects in which port to poison the ECC encoder. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
7	ococc_poison_egen_xpi_rd_out	R/W	Poisons the ECC encoder at the AXI read data interface. This will not cause any interrupt to flag. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3

Bits	Name	Memory Access	Description
6:2	ococc_poison_egen_mr_rd_0_byte_num	R/W	Byte number to poison for the AXI write data interface ECC encoder. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
1	ococc_poison_egen_mr_rd_0	R/W	Poisons the ECC encoder at the AXI write data interface. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3
0	ococc_poison_en	R/W	Enables poisoning of ECC encoders and parity generators. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3

1.2.82 OCECCSTAT0

- **Description:** On-Chip ECC Status Register 0
- **Size:** 32 bits
- **Offset:** 0x10708
- **Exists:** UMCTL2_OCECC_EN_1==1

This register is in block REGB_DDRC_CH0.

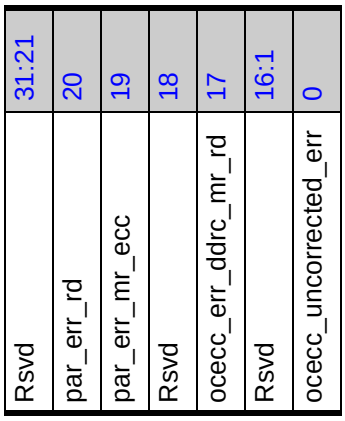


Table 1-141 Fields for Register: OCECCSTAT0

Bits	Name	Memory Access	Description
31:21			**Reserved Field:** Yes
20	par_err_rd	R	Parity error for read ECC data. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
19	par_err_mr_ecc	R	Parity error for write ECC data. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
18			**Reserved Field:** Yes
17	ocecc_err_ddrc_mr_rd	R	ECC error for write data RAM. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
16:1			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
0	ocecc_uncorrected_err	R	ECC uncorrected error or parity error detected in one of the ECC decoders/Parity checkers. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.2.83 OCECCSTAT1

- **Description:** On-Chip ECC Status Register 1
- **Size:** 32 bits
- **Offset:** 0x1070c
- **Exists:** UMCTL2_OCECC_EN_1==1

This register is in block REGB_DDRC_CH0.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ocecc_err_xpi_rd_15	ocecc_err_xpi_rd_14	ocecc_err_xpi_rd_13	ocecc_err_xpi_rd_12	ocecc_err_xpi_rd_11	ocecc_err_xpi_rd_10	ocecc_err_xpi_rd_9	ocecc_err_xpi_rd_8	ocecc_err_xpi_rd_7	ocecc_err_xpi_rd_6	ocecc_err_xpi_rd_5	ocecc_err_xpi_rd_4	ocecc_err_xpi_rd_3	ocecc_err_xpi_rd_2	ocecc_err_xpi_rd_1	ocecc_err_xpi_rd_0	ocecc_err_xpi_wr_in_15	ocecc_err_xpi_wr_in_14	ocecc_err_xpi_wr_in_13	ocecc_err_xpi_wr_in_12	ocecc_err_xpi_wr_in_11	ocecc_err_xpi_wr_in_10	ocecc_err_xpi_wr_in_9	ocecc_err_xpi_wr_in_8	ocecc_err_xpi_wr_in_7	ocecc_err_xpi_wr_in_6	ocecc_err_xpi_wr_in_5	ocecc_err_xpi_wr_in_4	ocecc_err_xpi_wr_in_3	ocecc_err_xpi_wr_in_2	ocecc_err_xpi_wr_in_1	ocecc_err_xpi_wr_in_0

Table 1-142 Fields for Register: OCECCSTAT1

Bits	Name	Memory Access	Description
31	ocecc_err_xpi_rd_15	R	OCECC read data error detected in AXI port 15. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_15==1 **Programming Mode:** Static
30	ocecc_err_xpi_rd_14	R	OCECC read data error detected in AXI port 14. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_14==1 **Programming Mode:** Static
29	ocecc_err_xpi_rd_13	R	OCECC read data error detected in AXI port 13. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_13==1 **Programming Mode:** Static
28	ocecc_err_xpi_rd_12	R	OCECC read data error detected in AXI port 12. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_12==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
27	ocecc_err_xpi_rd_11	R	OCECC read data error detected in AXI port 11. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_11==1 **Programming Mode:** Static
26	ocecc_err_xpi_rd_10	R	OCECC read data error detected in AXI port 10. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_10==1 **Programming Mode:** Static
25	ocecc_err_xpi_rd_9	R	OCECC read data error detected in AXI port 9. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_9==1 **Programming Mode:** Static
24	ocecc_err_xpi_rd_8	R	OCECC read data error detected in AXI port 8. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_8==1 **Programming Mode:** Static
23	ocecc_err_xpi_rd_7	R	OCECC read data error detected in AXI port 7. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_7==1 **Programming Mode:** Static
22	ocecc_err_xpi_rd_6	R	OCECC read data error detected in AXI port 6. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_6==1 **Programming Mode:** Static
21	ocecc_err_xpi_rd_5	R	OCECC read data error detected in AXI port 5. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_5==1 **Programming Mode:** Static
20	ocecc_err_xpi_rd_4	R	OCECC read data error detected in AXI port 4. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_4==1 **Programming Mode:** Static
19	ocecc_err_xpi_rd_3	R	OCECC read data error detected in AXI port 3. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_3==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
18	ocecc_err_xpi_rd_2	R	OCECC read data error detected in AXI port 2. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_2==1 **Programming Mode:** Static
17	ocecc_err_xpi_rd_1	R	OCECC read data error detected in AXI port 1. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_1==1 **Programming Mode:** Static
16	ocecc_err_xpi_rd_0	R	OCECC read data error detected in AXI port 0. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_0==1 **Programming Mode:** Static
15	ocecc_err_xpi_wr_in_15	R	OCECC write data error detected in AXI port 15. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_15==1 **Programming Mode:** Static
14	ocecc_err_xpi_wr_in_14	R	OCECC write data error detected in AXI port 14. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_14==1 **Programming Mode:** Static
13	ocecc_err_xpi_wr_in_13	R	OCECC write data error detected in AXI port 13. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_13==1 **Programming Mode:** Static
12	ocecc_err_xpi_wr_in_12	R	OCECC write data error detected in AXI port 12. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_12==1 **Programming Mode:** Static
11	ocecc_err_xpi_wr_in_11	R	OCECC write data error detected in AXI port 11. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_11==1 **Programming Mode:** Static
10	ocecc_err_xpi_wr_in_10	R	OCECC write data error detected in AXI port 10. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_10==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
9	ocecc_err_xpi_wr_in_9	R	OCECC write data error detected in AXI port 9. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_9==1 **Programming Mode:** Static
8	ocecc_err_xpi_wr_in_8	R	OCECC write data error detected in AXI port 8. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_8==1 **Programming Mode:** Static
7	ocecc_err_xpi_wr_in_7	R	OCECC write data error detected in AXI port 7. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_7==1 **Programming Mode:** Static
6	ocecc_err_xpi_wr_in_6	R	OCECC write data error detected in AXI port 6. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_6==1 **Programming Mode:** Static
5	ocecc_err_xpi_wr_in_5	R	OCECC write data error detected in AXI port 5. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_5==1 **Programming Mode:** Static
4	ocecc_err_xpi_wr_in_4	R	OCECC write data error detected in AXI port 4. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_4==1 **Programming Mode:** Static
3	ocecc_err_xpi_wr_in_3	R	OCECC write data error detected in AXI port 3. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_3==1 **Programming Mode:** Static
2	ocecc_err_xpi_wr_in_2	R	OCECC write data error detected in AXI port 2. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_2==1 **Programming Mode:** Static
1	ocecc_err_xpi_wr_in_1	R	OCECC write data error detected in AXI port 1. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_1==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
0	ocecc_err_xpi_wr_in_0	R	OCECC write data error detected in AXI port 0. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_AXI_0==1 **Programming Mode:** Static

1.2.84 OCECCSTAT2

- **Description:** On-Chip ECC Status Register 2
- **Size:** 32 bits
- **Offset:** 0x10710
- **Exists:** UMCTL2_OCECC_EN_1==1

This register is in block REGB_DDRC_CH0.

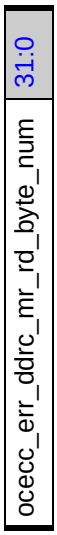


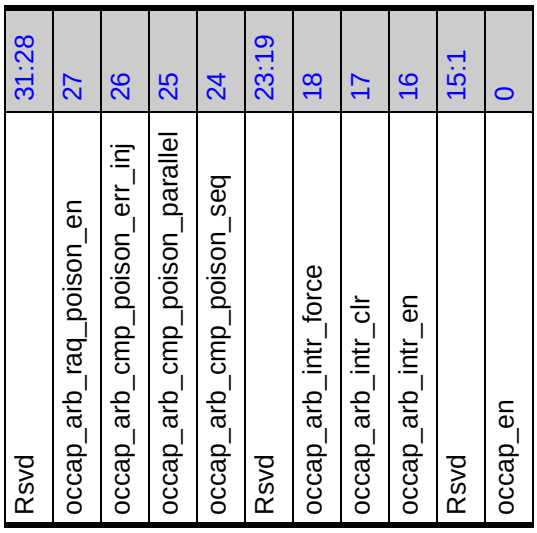
Table 1-143 Fields for Register: OCECCSTAT2

Bits	Name	Memory Access	Description
31:0	ococc_err_ddrc_mr_rd_byte_num	R	Byte index which caused an ECC error at the write data RAM. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.2.85 OCCAPCFG

- **Description:** On-Chip command/Address Protection Configuration Register
- **Size:** 32 bits
- **Offset:** 0x10780
- **Exists:** UMCTL2_OCCAP_EN_1==1

This register is in block REGB_DDRC_CH0.



31:28	27	26	25	24	23:19	18	17	16	15:1	0
Rsvd	occap_arb_raq_poison_en	occap_arb_cmp_poison_err_inj	occap_arb_cmp_poison_parallel	occap_arb_cmp_poison_seq	Rsvd	occap_arb_intr_force	occap_arb_intr_clr	occap_arb_intr_en	Rsvd	occap_en

Table 1-144 Fields for Register: OCCAPCFG

Bits	Name	Memory Access	Description
31:28			**Reserved Field:** Yes
27	occap_arb_raq_poison_en	R/W	Enables poisoning for the Read Address Queues (RAQ) inside each XPI. Poisoning inverts all parity bits generated by the parity generator. Error will be flagged as soon as the first RAQ is read. This register is not cleared automatically and must be reprogrammed to 0 at the end of the operation. **Value After Reset:** 0x0 **Exists:** UMCTL2_OCCAP_EN_1==1 && UMCTL2_INCL_ARB==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
26	occap_arb_cmp_poison_err_inj	R/W	Enable error injection in the poisoning of OCCAP Arbiter logic Injects error into poisoning logic (either parallel or seq) such that XOR logic for one signal is not poisoned when expected. If set, it allows ability to corrupt the following register fields. ■ 1'b0: OCCAPSTAT.occap_arb_cmp_poison_parallel/seq_err=0 ■ 1'b1: OCCAPSTAT.occap_arb_cmp_poison_parallel/seq_err=1 Do not change value in same APB write as setting of occap_arb_cmp_poison_parallel/_seq **Value After Reset:** 0x0 **Exists:** UMCTL2_OCCAP_EN_1==1 && UMCTL2_INCL_ARB==1 **Programming Mode:** Dynamic
25	occap_arb_cmp_poison_parallel	R/W1C	Enables full poisoning for compare logic inside XPI. Poisoning inverts all bits of all outputs coming from the duplicated modules before the XOR comparators together. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** UMCTL2_OCCAP_EN_1==1 && UMCTL2_INCL_ARB==1 **Testable:** readOnly **Programming Mode:** Dynamic
24	occap_arb_cmp_poison_seq	R/W1C	Enables poisoning for compare logic inside XPI. Poisoning inverts all bits coming from the duplicated modules before the XOR comparators one output at the time per each comparator. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** UMCTL2_OCCAP_EN_1==1 && UMCTL2_INCL_ARB==1 **Testable:** readOnly **Programming Mode:** Dynamic
23:19			**Reserved Field:** Yes
18	occap_arb_intr_force	R/W1C	Interrupt force bit for occap_arb_err_intr, setting this register will cause the output interrupt to be asserted. DDRCTL automatically clears this bit. There is no interaction between functionally triggering an interrupt and forcing an interrupt (they are mutually exclusive). **Value After Reset:** 0x0 **Exists:** UMCTL2_OCCAP_EN_1==1 && UMCTL2_INCL_ARB==1 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
17	occap_arb_intr_clr	R/W1C	Interrupt clear bit for occap_arb_err_intr and occap_arb_cmp_poison_complete. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** UMCTL2_OCCAP_EN_1==1 && UMCTL2_INCL_ARB==1 **Testable:** readOnly **Programming Mode:** Dynamic
16	occap_arb_intr_en	R/W	Enables interrupt generation upon detection of OCCAP Arbiter errors. **Value After Reset:** 0x1 **Exists:** UMCTL2_OCCAP_EN_1==1 && UMCTL2_INCL_ARB==1 **Programming Mode:** Dynamic
15:1			**Reserved Field:** Yes
0	occap_en	R/W	On Chip Command/Address Path Protection (OCCAP) enable register. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3

1.2.86 OCCAPSTAT

- **Description:** On-Chip command/Address Protection Status Register
- **Size:** 32 bits
- **Offset:** 0x10784
- **Exists:** UMCTL2_OCCAP_EN_1==1 && UMCTL2_INCL_ARB==1

This register is in block REGB_DDRC_CH0.

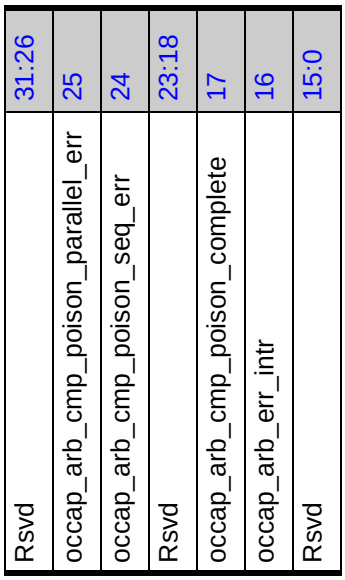


Table 1-145 Fields for Register: OCCAPSTAT

Bits	Name	Memory Access	Description
31:26			**Reserved Field:** Yes
25	occap_arb_cmp_poison_parallel_err	R	Error when occap_arb_cmp_poison_full_en was active due to incorrect no. of errors being occurring. Internal logic checks the number of errors detected while poisoning occurred for occap_arb_cmp_poison_full_en. It checks for error on all of the the corresponding XOR outputs. If multi-bit, checks also that all XOR bits are set. Register is valid when occap_arb_cmp_poison_complete=1. **Value After Reset:** 0x0 **Exists:** UMCTL2_OCCAP_EN_1==1 && UMCTL2_INCL_ARB==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
24	occap_arb_cmp_poison_seq_err	R	Error when occap_arb_cmp_poison_en was active due to incorrect no. of errors being occurring. Internal logic checks that the correct number of errors detected while poisoning one output at the time occurred for occap_arb_cmp_poison_en. It checks for error on one output at the time. Register is valid when occap_arb_cmp_poison_complete=1. **Value After Reset:** 0x0 **Exists:** UMCTL2_OCCAP_EN_1==1 && UMCTL2_INCL_ARB==1 **Programming Mode:** Static
23:18			**Reserved Field:** Yes
17	occap_arb_cmp_poison_complete	R	OCCAP ARB comparator poisoning complete interrupt status. Register cleared by OCCAPCFG.occap_arb_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_OCCAP_EN_1==1 && UMCTL2_INCL_ARB==1 **Programming Mode:** Static
16	occap_arb_err_intr	R	OCCAP Arbiter error interrupt status. Register cleared by OCCAPCFG.occap_arb_intr_clr. **Value After Reset:** 0x0 **Exists:** UMCTL2_OCCAP_EN_1==1 && UMCTL2_INCL_ARB==1 **Programming Mode:** Static
15:0			**Reserved Field:** Yes

1.2.87 OCCAPCFG1

- **Description:** On-Chip command/Address Protection Configuration Register 1
- **Size:** 32 bits
- **Offset:** 0x10788
- **Exists:** UMCTL2_OCCAP_EN_1==1

This register is in block REGB_DDRC_CH0.

31:27	26	25	24	23:19	18	17	16	15:11	10	9	8	7:3	2	1	0
Rsvd	occap_ddrc_ctrl_poison_err_inj	occap_ddrc_ctrl_poison_parallel	occap_ddrc_ctrl_poison_seq	Rsvd	occap_ddrc_ctrl_intr_force	occap_ddrc_ctrl_intr_clr	occap_ddrc_ctrl_intr_en	Rsvd	occap_ddrc_data_poison_err_inj	occap_ddrc_data_poison_parallel	occap_ddrc_data_poison_seq	Rsvd	occap_ddrc_data_intr_force	occap_ddrc_data_intr_clr	occap_ddrc_data_intr_en

Table 1-146 Fields for Register: OCCAPCFG1

Bits	Name	Memory Access	Description
31:27			**Reserved Field:** Yes
26	occap_ddrc_ctrl_poison_err_inj	R/W	Enable error injection in the poisoning of OCCAP DDRC CTRL logic Injects error into poisoning logic (either parallel or seq) such that XOR logic for one signal is not poisoned when expected. If set, it allows ability to corrupt the following register fields. ■ 1'b0: OCCAPSTAT1.occap_ddrc_ctrl_poison_parallel/seq_err=0 ■ 1'b1: OCCAPSTAT1.occap_ddrc_ctrl_poison_parallel/seq_err=1 Do not change value in same APB write as setting of occap_ddrc_ctrl_poison_parallel/_seq **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
25	occap_ddrc_ctrl_poison_parallel	R/W1C	Enables poisoning of OCCAP DDRC CTRL logic for all parts of comparison logic, in parallel. Poisons comparison logic for one core_ddrc_core_clk cycle by inverting a ddrc_ctrl[0]'s signal to XOR logic. ddrc_ctrl[1] related signals are never poisoned. All signals are poisoned in parallel and checks if all signals were poisoned correctly. If this is not the case, this is flagged by OCCAPSTAT1.occap_ddrc_ctrl_poison_parallel_err=1. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
24	occap_ddrc_ctrl_poison_seq	R/W1C	Enables poisoning of OCCAP DDRC CTRL logic for all parts of comparison logic, in sequence. Poisons comparison logic for one core_ddrc_core_clk cycle by inverting a ddrc_ctrl[0]'s signal to XOR logic. ddrc_ctrl[1] related signals are never poisoned. Each signal from ddrc_ctrl[0] is poisoned in series and checks in turn, that each signal was poisoned correctly. If this is not the case, this is flagged by OCCAPSTAT1.occap_ddrc_ctrl_poison_seq_err=1. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
23:19			**Reserved Field:** Yes
18	occap_ddrc_ctrl_intr_force	R/W1C	Interrupt force bit for occap_ddrc_ctrl_err_intr, setting this register will cause the output interrupt to be asserted. DDRCTL automatically clears this bit. There is no interaction between functionally triggering an interrupt and forcing an interrupt (they are mutually exclusive). **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
17	occap_ddrc_ctrl_intr_clr	R/W1C	Interrupt clear bit for occap_ddrc_ctrl_err_intr. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
16	occap_ddrc_ctrl_intr_en	R/W	Enables interrupt generation on signal occap_ddrc_ctrl_err_intr upon detection of OCCAP DDRC CTRL errors. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Dynamic
15:11			**Reserved Field:** Yes
10	occap_ddrc_data_poison_err_inj	R/W	Enable error injection in the poisoning of OCCAP DDRC DATA logic Injects error into poisoning logic (either parallel or seq) such that XOR logic for one signal is not poisoned when expected. If set, it allows ability to corrupt the following register fields. ■ 1'b0: OCCAPSTAT1.occap_ddrc_data_poison_parallel/seq_err=0 ■ 1'b1: OCCAPSTAT1.occap_ddrc_data_poison_parallel/seq_err=1 Do not change value in same APB write as setting of occap_ddrc_data_poison_parallel/_seq **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
9	occap_ddrc_data_poison_parallel	R/W1C	Enables poisoning of OCCAP DDRC DATA logic for all parts of comparison logic, in parallel. Poisons comparison logic for one core_ddrc_core_clk cycle by inverting all bits of a signal to XOR logic. All signals of instance[0] of the duplicated modules are poisoned in parallel and checks if all signals were poisoned correctly. If this is not the case, this is flagged by OCCAPSTAT1.occap_ddrc_data_poison_parallel_err=1. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
8	occap_ddrc_data_poison_seq	R/W1C	Enables poisoning of OCCAP DDRC DATA logic for all parts of comparison logic, in sequence. Poisons comparison logic for one core_ddrc_core_clk cycle by inverting all bits of a signal to XOR logic. All signals of instance[0] of the duplicated modules are poisoned in parallel and checks if all signals were poisoned correctly. If this is not the case, this is flagged by OCCAPSTAT1.occap_ddrc_data_poison_seq_err=1. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
7:3			**Reserved Field:** Yes
2	occap_ddrc_data_intr_force	R/W1C	Interrupt force bit for occap_ddrc_data_err_intr, setting this register will cause the output interrupt to be asserted. DDRCTL automatically clears this bit. There is no interaction between functionally triggering an interrupt and forcing an interrupt (they are mutually exclusive). **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
1	occap_ddrc_data_intr_clr	R/W1C	Interrupt clear bit for occap_ddrc_data_err_intr. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
0	occap_ddrc_data_intr_en	R/W	Enables interrupt generation on signal occap_ddrc_data_err_intr upon detection of OCCAP DDRC DATA errors. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Dynamic

1.2.88 OCCAPSTAT1

- **Description:** On-Chip command/Address Protection Status Register 1
- **Size:** 32 bits
- **Offset:** 0x1078c
- **Exists:** UMCTL2_OCCAP_EN_1==1

This register is in block REGB_DDRC_CH0.

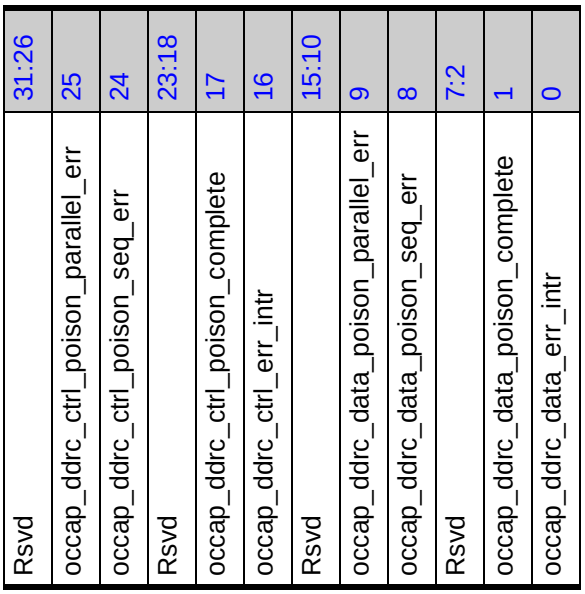


Table 1-147 Fields for Register: OCCAPSTAT1

Bits	Name	Memory Access	Description
31:26			**Reserved Field:** Yes
25	occap_ddrc_ctrl_poison_parallel_err	R	Error when occap_ddrc_ctrl_poison_parallel was active due to incorrect no. of errors being occurring. Internal logic checks the number of errors detected while poisoning occurred for occap_ddrc_ctrl_poison_parallel. It checks for error on all of the the corresponding XOR outputs. If multi-bit, checks also that all XOR bits are set. It checks all XOR in parallel. Register is valid only when occap_ddrc_ctrl_cmp_poison_complete=1. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

Bits	Name	Memory Access	Description
24	occap_ddrc_ctrl_poison_seq_err	R	Error when occap_ddrc_ctrl_poison_seq was active due to incorrect no. of errors being occurring. Internal logic checks the number of errors detected while poisoning occurred for occap_ddrc_ctrl_poison_seq. It checks for error on all of the corresponding XOR outputs. It checks each XOR sequentially. Register is valid only when occap_ddrc_ctrl_cmp_poison_complete=1. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
23:18			**Reserved Field:** Yes
17	occap_ddrc_ctrl_poison_complete	R	OCCAP DDRC CTRL poisoning complete interrupt status. Register cleared by OCCAPCFG1.occap_ddrc_ctrl_err_intr_clr. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
16	occap_ddrc_ctrl_err_intr	R	OCCAP DDRC CTRL error interrupt status. Register cleared by OCCAPCFG1.occap_ddrc_ctrl_err_intr_clr. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
15:10			**Reserved Field:** Yes
9	occap_ddrc_data_poison_parallel_err	R	Error when occap_ddrc_data_poison_parallel was active due to incorrect no. of errors being occurring. Internal logic checks the number of errors detected while poisoning occurred for occap_ddrc_data_poison_parallel. It checks for error on all of the the corresponding XOR outputs. If multi-bit, checks also that all XOR bits are set. It checks all XOR in parallel. This is cleared when OCCAPCFG1.occap_ddrc_data_poison_parallel=0 occurs. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

Bits	Name	Memory Access	Description
8	occap_ddrc_data_poison_seq_err	R	Error when occap_ddrc_data_poison_seq was active due to incorrect no. of errors being occurring. Internal logic checks the number of errors detected while poisoning occurred for occap_ddrc_data_poison_seq. It checks for error on all of the corresponding XOR outputs. It checks each XOR sequentially. This is cleared when OCCAPCFG1.occap_ddrc_data_poison_seq=0 occurs. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:2			**Reserved Field:** Yes
1	occap_ddrc_data_poison_complete	R	OCCAP DDRC DATA poisoning complete interrupt status. Register cleared by OCCAPCFG1.occap_ddrc_data_err_intr_clr. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
0	occap_ddrc_data_err_intr	R	OCCAP DDRC DATA error interrupt status. Register cleared by OCCAPCFG1.occap_ddrc_data_err_intr_clr. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.2.89 OCCAPCFG2

- **Description:** On-Chip command/Address Protection Configuration Register 2
- **Size:** 32 bits
- **Offset:** 0x10790
- **Exists:** UMCTL2_OCCAP_EN_1==1

This register is in block REGB_DDRC_CH0.

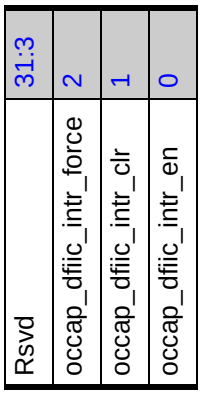


Table 1-148 Fields for Register: OCCAPCFG2

Bits	Name	Memory Access	Description
31:3			**Reserved Field:** Yes
2	occap_dfiic_intr_force	R/W1C	Interrupt force bit for occap_dfiic_err_intr, setting this register will cause the output interrupt to be asserted. DDRCTL automatically clears this bit. There is no interaction between functionally triggering an interrupt and forcing an interrupt (they are mutually exclusive). **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
1	occap_dfiic_intr_clr	R/W1C	Interrupt clear bit for occap_dfiic_err_intr. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
0	occap_dfiic_intr_en	R/W	Enables interrupt generation on signal occap_dfiic_err_intr upon detection of OCCAP DFI interconnect errors. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Dynamic

1.2.90 OCCAPSTAT2

- **Description:** On-Chip command/Address Protection Status Register 2
- **Size:** 32 bits
- **Offset:** 0x10794
- **Exists:** UMCTL2_OCCAP_EN_1==1

This register is in block REGB_DDRC_CH0.

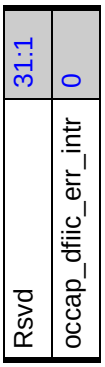


Table 1-149 Fields for Register: OCCAPSTAT2

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	occap_dfiic_err_intr	R	OCCAP DFI interconnect error interrupt status. Register cleared by OCCAPCFG2.occap_dfiic_intr_clr. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.2.91 REGPARCFG

- **Description:** Register Parity Configuration Register
- **Size:** 32 bits
- **Offset:** 0x10880
- **Exists:** UMCTL2_REGPAR_EN_1

This register is in block REGB_DDRC_CH0.
Note: all fields must be programmed with single write operation

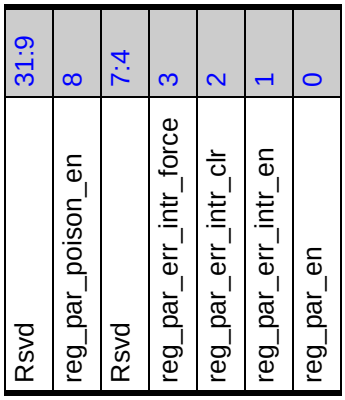


Table 1-150 Fields for Register: REGPARCFG

Bits	Name	Memory Access	Description
31:9			**Reserved Field:** Yes
8	reg_par_poison_en	R/W	Enable Register Parity poisoning. **Value After Reset:** 0x0 **Exists:** UMCTL2_REGPAR_EN_1 **Programming Mode:** Dynamic
7:4			**Reserved Field:** Yes
3	reg_par_err_intr_force	R/W1C	Interrupt force bit for reg_par_err_intr, setting this register will cause the output interrupt to be asserted. DDRCTL automatically clears this bit. There is no interaction between functionally triggering an interrupt and forcing an interrupt (they are mutually exclusive). **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
2	reg_par_err_intr_clr	R/W1C	Interupt clear bit for reg_par_err_intr. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** UMCTL2_REGPAR_EN_1 **Testable:** readOnly **Programming Mode:** Dynamic
1	reg_par_err_intr_en	R/W	Enables interrupt generation, if set to 1, on signal reg_par_err_intr upon detection of register parity error. **Value After Reset:** 0x1 **Exists:** UMCTL2_REGPAR_EN_1 **Programming Mode:** Dynamic
0	reg_par_en	R/W	Register Parity enable register. **Value After Reset:** 0x0 **Exists:** UMCTL2_REGPAR_EN_1 **Programming Mode:** Dynamic

1.2.92 REGPARSTAT

- **Description:** Register Parity Status Register
- **Size:** 32 bits
- **Offset:** 0x10884
- **Exists:** UMCTL2_REGPAR_EN_1

This register is in block REGB_DDRC_CH0.

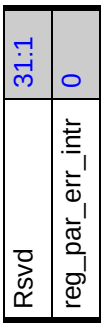


Table 1-151 Fields for Register: REGPARSTAT

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	reg_par_err_intr	R	Interrupt asserted when Register Parity error is detected. Cleared by setting REGPARCFG.reg_par_err_intr_clr to 1. **Value After Reset:** 0x0 **Exists:** UMCTL2_REGPAR_EN_1 **Programming Mode:** Static

1.2.93 LNKECCCTL0

- **Description:** Link-ECC Control Register 0
- **Size:** 32 bits
- **Offset:** 0x10980
- **Exists:** MEMC_LINK_ECC==1

This register is in block REGB_DDRC_CH0.

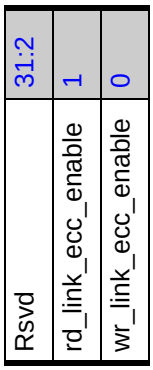


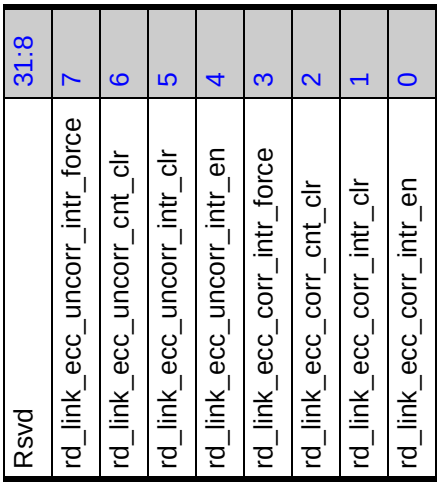
Table 1-152 Fields for Register: LNKECCCTL0

Bits	Name	Memory Access	Description
31:2			**Reserved Field:** Yes
1	rd_link_ecc_enable	R/W	Enable LPDDR5 Read Link ECC feature. ■ 0 - Disabel LPDDR5 Read Link ECC ■ 1 - Enable LPDDR5 Read Link ECC When non-LPDDR5 devices are used, this register must be set to 0. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Volatile:** true **Programming Mode:** Static
0	wr_link_ecc_enable	R/W	Enable LPDDR5 Write Link ECC feature. ■ 0 - Disabel LPDDR5 Write Link ECC ■ 1 - Enable LPDDR5 Write Link ECC When non-LPDDR5 devices are used, this register must be set to 0. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Volatile:** true **Programming Mode:** Static

1.2.94 LNKECCCTL1

- **Description:** Link-ECC Control Register 1
- **Size:** 32 bits
- **Offset:** 0x10984
- **Exists:** MEMC_LINK_ECC==1

This register is in block REGB_DDRC_CH0.
Note: Do not perform any APB access to LNKECCCTL1 within 32 pclk cycles of previous access to LNKECCCTL1, as this might lead to data loss.



31:8	7	6	5	4	3	2	1	0
Rsvd	rd_link_ecc_uncorr_intr_force	rd_link_ecc_uncorr_cnt_clr	rd_link_ecc_uncorr_intr_clr	rd_link_ecc_uncorr_intr_en	rd_link_ecc_corr_intr_force	rd_link_ecc_corr_cnt_clr	rd_link_ecc_corr_intr_clr	rd_link_ecc_corr_intr_en

Table 1-153 Fields for Register: LNKECCCTL1

Bits	Name	Memory Access	Description
31:8			**Reserved Field:** Yes
7	rd_link_ecc_uncorr_intr_force	R/W1C	Interrupt force bit for rd_linkecc_uncorr_err_intr. Setting this register will cause the output interrupt to be asserted. DDRCTL automatically clears this bit. There is no interaction between functionally triggering an interrupt and forcing an interrupt (they are mutually exclusive). **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Testable:** readOnly **Programming Mode:** Dynamic
6	rd_link_ecc_uncorr_cnt_clr	R/W1C	Clear all Read Link-ECC uncorrectable error count. If this bit set,LNKECCERRCNT0.rd_link_ecc_uncorr_cnt will be cleared. LPDDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
5	rd_link_ecc_uncorr_intr_clr	R/W1C	Clear Read Link-ECC uncorrectable error interrupt. If this bit set, rd_linkecc_uncorr_err_intr will be cleared. LPDDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Testable:** readOnly **Programming Mode:** Dynamic
4	rd_link_ecc_uncorr_intr_en	R/W	Interrupt enable bit for Read Link-ECC uncorrectable error. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic
3	rd_link_ecc_corr_intr_force	R/W1C	Interrupt force bit for rd_linkecc_corr_err_intr. Setting this register will cause the output interrupt to be asserted. DDRCTL automatically clears this bit. There is no interaction between functionally triggering an interrupt and forcing an interrupt (they are mutually exclusive). **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Testable:** readOnly **Programming Mode:** Dynamic
2	rd_link_ecc_corr_cnt_clr	R/W1C	Clear all Read Link-ECC correctable error count. If this bit set,LNKECCERRCNT0.rd_link_ecc_corr_cnt will be cleared. LPDDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Testable:** readOnly **Programming Mode:** Dynamic
1	rd_link_ecc_corr_intr_clr	R/W1C	Clear Read Link-ECC correctable error interrupt. If this bit set, rd_linkecc_corr_err_intr will be cleared. LPDDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
0	rd_link_ecc_corr_intr_en	R/W	Interrupt enable bit for Read Link-ECC correctable error. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic

1.2.95 LNKECCPOISONCTL0

- **Description:** Link-ECC Poison Control Register 0
- **Size:** 32 bits
- **Offset:** 0x10988
- **Exists:** MEMC_LINK_ECC==1

This register is in block REGB_DDRC_CH0.

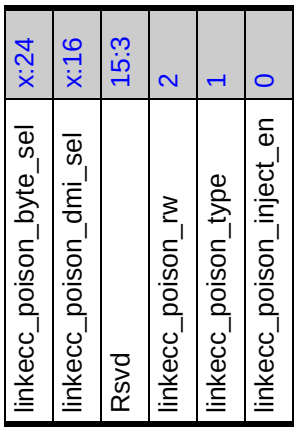


Table 1-154 Fields for Register: LNKECCPOISONCTL0

Bits	Name	Memory Access	Description
x:24	linkecc_poison_byte_sel	R/W	Select target byte(s) of Data for Read/Write Link ECC poisoning. This is bit map indicator. Bit N corresponding to Data[N*8+:8]. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic **RangeVariable[x]:**"MEMC_DRAM_TOTAL_DATA_WIDTH/8" + 23
x:16	linkecc_poison_dmi_sel	R/W	Select target DMI(s) of Data for Write Link ECC poisoning. This is bit map indicator. Bit N corresponding to DMI[N]. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic **RangeVariable[x]:**"MEMC_DRAM_TOTAL_DATA_WIDTH/8" + 15
15:3			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
2	linkecc_poison_rw	R/W	Indicates whether the Link-ECC poisoning operation is Read or Write. ■ 0 - Write ■ 1 - Read **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic
1	linkecc_poison_type	R/W	Indicates whether the Link-ECC poisoning operation is Single-bit error or Double bit error. ■ 0 - Single bit Error ■ 1 - Double bit Error **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic
0	linkecc_poison_inject_en	R/W	Setting this register bit to 1 triggers the Link-ECC poisoning. Once Link-ECC is poisoned to a ECC code, the ECC poisoning is completed automatically and LNKECCPOISONSTAT.linkecc_poison_complete becomes 1. Please make sure that LNKECCPOISONSTAT.linkecc_poison_complete==0 before writing this register to 1. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic

1.2.96 LNKECCPOISONSTAT

- **Description:** Link-ECC Poison Status Register
- **Size:** 32 bits
- **Offset:** 0x1098c
- **Exists:** MEMC_LINK_ECC==1

This register is in block REGB_DDRC_CH0.

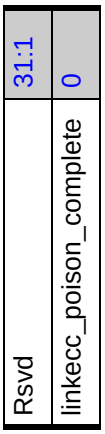


Table 1-155 Fields for Register: LNKECCPOISONSTAT

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	linkecc_poison_complete	R	Indicates Link-ECC poisoning operation is done. ■ 0 - Link-ECC poisoning is not completed ■ 1 - Link-ECC poisoning is completed **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic

1.2.97 LNKECCINDEX

- **Description:** Link-ECC Index Register
- **Size:** 32 bits
- **Offset:** 0x10990
- **Exists:** MEMC_LINK_ECC==1

This register is in block REGB_DDRC_CH0.

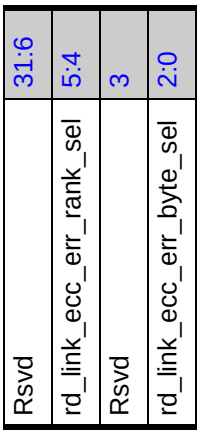


Table 1-156 Fields for Register: LNKECCINDEX

Bits	Name	Memory Access	Description
31:6			**Reserved Field:** Yes
5:4	rd_link_ecc_err_rank_sel	R/W	Select of which rank status output to LNKECCERRCNT.rd_link_ecc_uncorr_cnt, rd_link_ecc_corr_cnt and rd_link_ecc_err_syndrome. The value must be less than MEMC_NUM_RANKS. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 && MEMC_NUM_RANKS_GT_1==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1
3			**Reserved Field:** Yes
2:0	rd_link_ecc_err_byte_sel	R/W	Select of which data byte status output to LNKECCERRCNT.rd_link_ecc_uncorr_cnt, rd_link_ecc_corr_cnt and rd_link_ecc_err_syndrome. The value must be less than MEMC_DRAM_DATA_WIDTH/8. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1

1.2.98 LNKECCERRCNT0

- **Description:** Link-ECC Error Status Register 0
- **Size:** 32 bits
- **Offset:** 0x10994
- **Exists:** MEMC_LINK_ECC==1

This register is in block REGB_DDRC_CH0.

31:24	23:16	15:9	8:0
rd_link_ecc_uncorr_cnt	rd_link_ecc_corr_cnt	Rsvd	rd_link_ecc_err_syndrome

Table 1-157 Fields for Register: LNKECCERRCNT0

Bits	Name	Memory Access	Description
31:24	rd_link_ecc_uncorr_cnt	R	Indicates double bit error count. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic
23:16	rd_link_ecc_corr_cnt	R	Indicates single bit error count. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic
15:9			**Reserved Field:** Yes
8:0	rd_link_ecc_err_syndrome	R	Indicates ECC syndrome from most recent single bit error. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic

1.2.99 LNKECCERRSTAT

- **Description:** Link-ECC Error Status Register 1
- **Size:** 32 bits
- **Offset:** 0x10998
- **Exists:** MEMC_LINK_ECC==1

This register is in block REGB_DDRC_CH0.

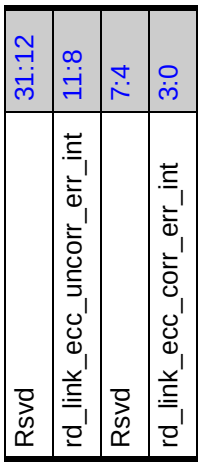


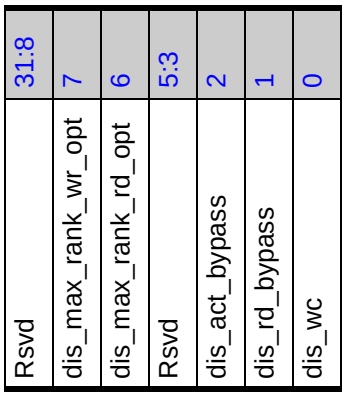
Table 1-158 Fields for Register: LNKECCERRSTAT

Bits	Name	Memory Access	Description
31:12			**Reserved Field:** Yes
11:8	rd_link_ecc_uncorr_err_int	R	Indicates double bit error for Read Link-ECC. If double bit error happens, this interrupt bit is set. It remains set until cleared by LNKECCCTL1.rd_link_ecc_uncorr_intr_clr. Each bit represents one rank. (LSB is the lowest rank number.) **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Static
7:4			**Reserved Field:** Yes
3:0	rd_link_ecc_corr_err_int	R	Indicates single bit error for Read Link-ECC. If signle bit error happens, this interrupt bit is set. It remains set until cleared by LNKECCCTL1.rd_link_ecc_corr_intr_clr. Each bit represents one rank. (LSB is the lowest rank number.) **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Static

1.2.100 OPCTRL0

- **Description:** Operation Control Register 0
- **Size:** 32 bits
- **Offset:** 0x10b80
- **Exists:** Always

This register is in block REGB_DDRC_CH0.



31:8	7	6	5:3	2	1	0
Rsvd	dis_max_rank_wr_opt	dis_max_rank_rd_opt	Rsvd	dis_act_bypass	dis_rd_bypass	dis_wc

Table 1-159 Fields for Register: OPCTRL0

Bits	Name	Memory Access	Description
31:8			**Reserved Field:** Yes
7	dis_max_rank_wr_opt	R/W	Disable optimized max_rank_wr and max_logical_rank_wr feature. This register is debug purpose only. For normal operation, This register must be set to 0. **Value After Reset:** 0x0 **Exists:** MEMC_NUM_RANKS_GT_1_OR_DDRCTL_DDRC_CID_WIDTH_GT_0 **Volatile:** true **Programming Mode:** Static
6	dis_max_rank_rd_opt	R/W	Disable optimized max_rank_rd and max_logical_rank_rd feature. This register is debug purpose only. For normal operation, This register must be set to 0. **Value After Reset:** 0x0 **Exists:** MEMC_NUM_RANKS_GT_1_OR_DDRCTL_DDRC_CID_WIDTH_GT_0 **Volatile:** true **Programming Mode:** Static
5:3			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
2	dis_act_bypass	R/W	Only present in designs supporting activate bypass. When 1, disable bypass path for high priority read activates FOR DEBUG ONLY. **Value After Reset:** 0x0 **Exists:** MEMC_BYPASS==1 **Programming Mode:** Static
1	dis_rd_bypass	R/W	Only present in designs supporting read bypass. When 1, disable bypass path for high priority read page hits FOR DEBUG ONLY. **Value After Reset:** 0x0 **Exists:** MEMC_BYPASS==1 **Programming Mode:** Static
0	dis_wc	R/W	When 1, disable write combine. FOR DEBUG ONLY **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.2.101 OPCTRL1

- **Description:** Operation Control Register 1
- **Size:** 32 bits
- **Offset:** 0x10b84
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

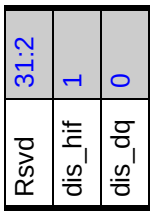


Table 1-160 Fields for Register: OPCTRL1

Bits	Name	Memory Access	Description
31:2			**Reserved Field:** Yes
1	dis_hif	R/W	When 1, DDRCTL asserts the HIF command signal hif_cmd_stall. DDRCTL will ignore the hif_cmd_valid and all other associated request signals. This bit is intended to be switched on-the-fly. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
0	dis_dq	R/W	When 1, DDRCTL will not de-queue any transactions from the CAM. Bypass is also disabled. All transactions are queued in the CAM. No reads or writes are issued to SDRAM as long as this is asserted. This bit may be used to prevent reads or writes being issued by the DDRCTL, which makes it safe to modify certain register fields associated with reads and writes (see Programming Chapter for details). After setting this bit, it is strongly recommended to poll OPCTRLCAM.wr_data_pipeline_empty and OPCTRLCAM.rd_data_pipeline_empty, before making changes to any registers which affect reads and writes. This will ensure that the relevant logic in the DDRC is idle. This bit is intended to be switched on-the-fly. Note: This bit is not applicable for designs working in DDR5 mode. In DDR5 mode, use software command interface command DisDqRef to achieve the same function as this bit. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.2.102 OPCTRLCAM

- **Description:** CAM Operation Control Register
- **Size:** 32 bits
- **Offset:** 0x10b88
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

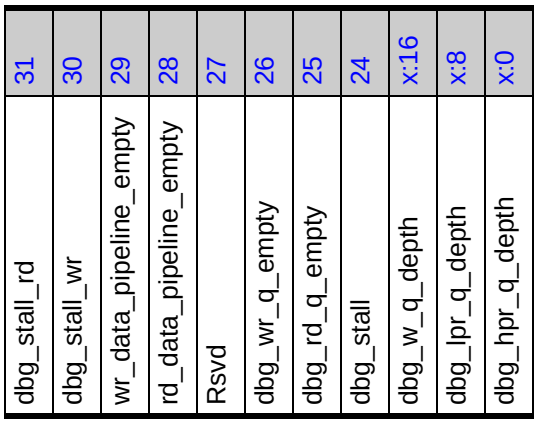


Table 1-161 Fields for Register: OPCTRLCAM

Bits	Name	Memory Access	Description
31	dbg_stall_rd	R	Stall for Read channel FOR DEBUG ONLY **Value After Reset:** 0x0 **Exists:** UMCTL2_DUAL_HIF_1==1 **Programming Mode:** Dynamic
30	dbg_stall_wr	R	Stall for Write channel FOR DEBUG ONLY **Value After Reset:** 0x0 **Exists:** UMCTL2_DUAL_HIF_1==1 **Programming Mode:** Dynamic
29	wr_data_pipeline_empty	R	This bit indicates that the write data pipeline on the DFI interface is empty. This register is intended to be polled at least twice after setting OPCTRL1.dis_dq, to ensure that all remaining commands/data have completed. **Value After Reset:** 0x0 **Exists:** Always **Reset Mask:** 0x0 **Volatile:** true **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
28	rd_data_pipeline_empty	R	This bit indicates that the read data pipeline on the DFI interface is empty. This register is intended to be polled at least twice after setting OPCTRL1.dis_dq, to ensure that all remaining commands/data have completed. **Value After Reset:** 0x0 **Exists:** Always **Reset Mask:** 0x0 **Volatile:** true **Programming Mode:** Dynamic
27			**Reserved Field:** Yes
26	dbg_wr_q_empty	R	When 1, all the Write command queues and Write data buffers inside DDRC are empty. This register is to be used for debug purpose. An example use-case scenario: When Controller enters Self-Refresh using the Low-Power entry sequence, Controller is expected to have executed all the commands in its queues and the write and read data drained. Hence this register must be 1 at that time. **Value After Reset:** 0x0 **Exists:** Always **Reset Mask:** 0x0 **Volatile:** true **Programming Mode:** Dynamic
25	dbg_rd_q_empty	R	When 1, all the Read command queues and Read data buffers inside DDRC are empty. This register is to be used for debug purpose. An example use-case scenario: When Controller enters Self-Refresh using the Low-Power entry sequence, Controller is expected to have executed all the commands in its queues and the write and read data drained. Hence this register must be 1 at that time. **Value After Reset:** 0x0 **Exists:** Always **Reset Mask:** 0x0 **Volatile:** true **Programming Mode:** Dynamic
24	dbg_stall	R	Stall FOR DEBUG ONLY **Value After Reset:** 0x0 **Exists:** UMCTL2_DUAL_HIF_1==0 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
x:16	dbg_w_q_depth	R	Write queue depth The last entry of WR queue is reserved for ECC SCRUB operation. This entry is not included in the calculation of the queue depth. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_WRCMD_ENTRY_BITS + 1" + 15
x:8	dbg_lpr_q_depth	R	Low priority read queue depth The last entry of Lpr queue is reserved for ECC SCRUB operation. This entry is not included in the calculation of the queue depth. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_RDCMD_ENTRY_BITS + 1" + 7
x:0	dbg_hpr_q_depth	R	High priority read queue depth **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_RDCMD_ENTRY_BITS + 1" - 1

1.2.103 OPCTRLCMD

- **Description:** Command Operation Control Register
- **Size:** 32 bits
- **Offset:** 0x10b8c
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

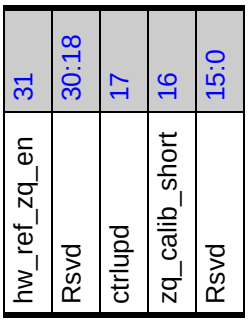


Table 1-162 Fields for Register: OPCTRLCMD

Bits	Name	Memory Access	Description
31	hw_ref_zq_en	R/W	Setting this register bit to 1 allows refresh and ZQCS/MPC(ZQ Calibration) commands to be triggered from hardware via the IOs ext_*. If set to 1, the fields OPCTRLCMD.zq_calib_short and OPREFCTRL*.rank*_refresh have no function, and are ignored by the DDRCTL logic. Setting this register bit to 0 allows refresh and ZQCS/MPC(ZQ Calibration) to be triggered from software, via the fields OPCTRLCMD.zq_calib_short and OPREFCTRL*.rank*_refresh. If set to 0, the hardware pins ext_* have no function, and are ignored by the DDRCTL logic. This register is static, and may only be changed when the DDRC reset signal, core_ddrc_rstn, is asserted (0). Note: Supporting this register field in this release is limited. Contact Synopsys if you wish to use this. Note: This field is not applicable for DDR5 ZQ Calibration. **Value After Reset:** 0x0 **Exists:** UMCTL2_REF_ZQ_IO==1 **Programming Mode:** Static
30:18			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
17	ctrlupd	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a dfi_ctrlupd_req to the PHY. When this request is stored in the DDRCTL, the bit is automatically cleared. This operation must only be performed when DFIUPD0.dis_auto_ctrlupd=1. Note: This field is not applicable for DDR5. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
16	zq_calib_short	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a ZQCS (ZQ calibration short)/MPC(ZQ calibration) command to the SDRAM. When this request is stored in the DDRCTL, the bit is automatically cleared. This operation can be performed only when ZQCTL0.dis_auto_zq=1. It is recommended NOT to set this register bit if in Init, in Self-Refresh(except LPDDR4/5) or SR-Powerdown(LPDDR4/5) or Deep Sleep Mode or Maximum Power Saving Mode. For Self-Refresh(except LPDDR4/5) or SR-Powerdown(LPDDR4/5) it will be scheduled after SR(except LPDDR4/5) or SRPD(LPDDR4/5) has been exited. For Deep Sleep Mode, it will be scheduled after DSM and/or SRPD has been exited. For Maximum Power Saving Mode, it will not be scheduled, although OPCTRLSTAT.zq_calib_short_busy will be de-asserted. Note: This field is not applicable for DDR5. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
15:0			**Reserved Field:** Yes

1.2.104 OPCTRLSTAT

- **Description:** Status Operation Control Register
- **Size:** 32 bits
- **Offset:** 0x10b90
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

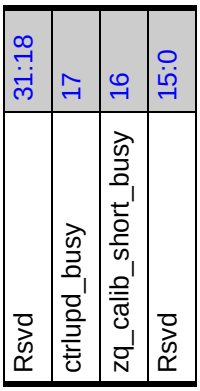


Table 1-163 Fields for Register: OPCTRLSTAT

Bits	Name	Memory Access	Description
31:18			**Reserved Field:** Yes
17	ctrlupd_busy	R	SoC core may initiate a ctrlupd operation only if this signal is low. This signal goes high in the clock after the DDRCTL accepts the ctrlupd request. It goes low when the ctrlupd operation is initiated in the DDRCTL. It is recommended not to perform ctrlupd operations when this signal is high. ■ 0 - Indicates that the SoC core can initiate a ctrlupd operation ■ 1 - Indicates that ctrlupd operation has not been initiated yet in the DDRCTL Note: This field is not applicable for DDR5. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
16	zq_calib_short_busy	R	SoC core may initiate a ZQCS (ZQ calibration short) operation only if this signal is low. This signal goes high in the clock after the DDRCTL accepts the ZQCS request. It goes low when the ZQCS operation is initiated in the DDRCTL. It is recommended not to perform ZQCS operations when this signal is high. ■ 0 - Indicates that the SoC core can initiate a ZQCS operation ■ 1 - Indicates that ZQCS operation has not been initiated yet in the DDRCTL Note: This field is not applicable for DDR5. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
15:0			**Reserved Field:** Yes

1.2.105 OPCTRLCAM1

- **Description:** CAM Operation Control Register 1
- **Size:** 32 bits
- **Offset:** 0x10b94
- **Exists:** MEMC_INLINE_ECC==1

This register is in block REGB_DDRC_CH0.

Table 1-164 Fields for Register: OPCTRLCAM1

Bits	Name	Memory Access	Description
x:0	dbg_wrecc_q_depth	R	Write ECC queue depth FOR DEBUG ONLY **Value After Reset:** 0x0 **Exists:** MEMC_INLINE_ECC==1 **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_WRCMD_ENTRY_BITS + 1" - 1

1.2.106 OPREFCTRL0

- **Description:** Refresh Operation Control Register 0
- **Size:** 32 bits
- **Offset:** 0x10b98
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
rank31_refresh	rank30_refresh	rank29_refresh	rank28_refresh	rank27_refresh	rank26_refresh	rank25_refresh	rank24_refresh	rank23_refresh	rank22_refresh	rank21_refresh	rank20_refresh	rank19_refresh	rank18_refresh	rank17_refresh	rank16_refresh	rank15_refresh	rank14_refresh	rank13_refresh	rank12_refresh	rank11_refresh	rank10_refresh	rank9_refresh	rank8_refresh	rank7_refresh	rank6_refresh	rank5_refresh	rank4_refresh	rank3_refresh	rank2_refresh	rank1_refresh	rank0_refresh

Table 1-165 Fields for Register: OPREFCTRL0

Bits	Name	Memory Access	Description
31	rank31_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 31. Writing to this bit causes OPREFSTAT0.rank31_refresh_busy to be set. When OPREFSTAT0.rank31_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 31. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>31 **Testable:** readOnly **Programming Mode:** Dynamic
30	rank30_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 30. Writing to this bit causes OPREFSTAT0.rank30_refresh_busy to be set. When OPREFSTAT0.rank30_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 30. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>30 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
29	rank29_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 29. Writing to this bit causes OPREFSTAT0.rank29_refresh_busy to be set. When OPREFSTAT0.rank29_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 29. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>29 **Testable:** readOnly **Programming Mode:** Dynamic
28	rank28_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 28. Writing to this bit causes OPREFSTAT0.rank28_refresh_busy to be set. When OPREFSTAT0.rank28_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 28. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>28 **Testable:** readOnly **Programming Mode:** Dynamic
27	rank27_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 27. Writing to this bit causes OPREFSTAT0.rank27_refresh_busy to be set. When OPREFSTAT0.rank27_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 27. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>27 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
26	rank26_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 26. Writing to this bit causes OPREFSTAT0.rank26_refresh_busy to be set. When OPREFSTAT0.rank26_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 26. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>26 **Testable:** readOnly **Programming Mode:** Dynamic
25	rank25_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 25. Writing to this bit causes OPREFSTAT0.rank25_refresh_busy to be set. When OPREFSTAT0.rank25_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 25. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>25 **Testable:** readOnly **Programming Mode:** Dynamic
24	rank24_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 24. Writing to this bit causes OPREFSTAT0.rank24_refresh_busy to be set. When OPREFSTAT0.rank24_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 24. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>24 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
23	rank23_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 23. Writing to this bit causes OPREFSTAT0.rank23_refresh_busy to be set. When OPREFSTAT0.rank23_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 23. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>23 **Testable:** readOnly **Programming Mode:** Dynamic
22	rank22_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 22. Writing to this bit causes OPREFSTAT0.rank22_refresh_busy to be set. When OPREFSTAT0.rank22_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 22. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>22 **Testable:** readOnly **Programming Mode:** Dynamic
21	rank21_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 21. Writing to this bit causes OPREFSTAT0.rank21_refresh_busy to be set. When OPREFSTAT0.rank21_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 21. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>21 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
20	rank20_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 20. Writing to this bit causes OPREFSTAT0.rank20_refresh_busy to be set. When OPREFSTAT0.rank20_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 20. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>20 **Testable:** readOnly **Programming Mode:** Dynamic
19	rank19_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 19. Writing to this bit causes OPREFSTAT0.rank19_refresh_busy to be set. When OPREFSTAT0.rank19_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 19. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>19 **Testable:** readOnly **Programming Mode:** Dynamic
18	rank18_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 18. Writing to this bit causes OPREFSTAT0.rank18_refresh_busy to be set. When OPREFSTAT0.rank18_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 18. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>18 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
17	rank17_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 17. Writing to this bit causes OPREFSTAT0.rank17_refresh_busy to be set. When OPREFSTAT0.rank17_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 17. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>17 **Testable:** readOnly **Programming Mode:** Dynamic
16	rank16_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 16. Writing to this bit causes OPREFSTAT0.rank16_refresh_busy to be set. When OPREFSTAT0.rank16_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 16. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>16 **Testable:** readOnly **Programming Mode:** Dynamic
15	rank15_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 15. Writing to this bit causes OPREFSTAT0.rank15_refresh_busy to be set. When OPREFSTAT0.rank15_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 15. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>15 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
14	rank14_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 14. Writing to this bit causes OPREFSTAT0.rank14_refresh_busy to be set. When OPREFSTAT0.rank14_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 14. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>14 **Testable:** readOnly **Programming Mode:** Dynamic
13	rank13_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 13. Writing to this bit causes OPREFSTAT0.rank13_refresh_busy to be set. When OPREFSTAT0.rank13_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 13. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>13 **Testable:** readOnly **Programming Mode:** Dynamic
12	rank12_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 12. Writing to this bit causes OPREFSTAT0.rank12_refresh_busy to be set. When OPREFSTAT0.rank12_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 12. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>12 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
11	rank11_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 11. Writing to this bit causes OPREFSTAT0.rank11_refresh_busy to be set. When OPREFSTAT0.rank11_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 11. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>11 **Testable:** readOnly **Programming Mode:** Dynamic
10	rank10_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 10. Writing to this bit causes OPREFSTAT0.rank10_refresh_busy to be set. When OPREFSTAT0.rank10_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 10. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>10 **Testable:** readOnly **Programming Mode:** Dynamic
9	rank9_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 9. Writing to this bit causes OPREFSTAT0.rank9_refresh_busy to be set. When OPREFSTAT0.rank9_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 9. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>9 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
8	rank8_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 8. Writing to this bit causes OPREFSTAT0.rank8_refresh_busy to be set. When OPREFSTAT0.rank8_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 8. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>8 **Testable:** readOnly **Programming Mode:** Dynamic
7	rank7_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 7. Writing to this bit causes OPREFSTAT0.rank7_refresh_busy to be set. When OPREFSTAT0.rank7_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 7. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>7 **Testable:** readOnly **Programming Mode:** Dynamic
6	rank6_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 6. Writing to this bit causes OPREFSTAT0.rank6_refresh_busy to be set. When OPREFSTAT0.rank6_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 6. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>6 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
5	rank5_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 5. Writing to this bit causes OPREFSTAT0.rank5_refresh_busy to be set. When OPREFSTAT0.rank5_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 5. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>5 **Testable:** readOnly **Programming Mode:** Dynamic
4	rank4_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 4. Writing to this bit causes OPREFSTAT0.rank4_refresh_busy to be set. When OPREFSTAT0.rank4_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 4. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>4 **Testable:** readOnly **Programming Mode:** Dynamic
3	rank3_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 3. Writing to this bit causes OPREFSTAT0.rank3_refresh_busy to be set. When OPREFSTAT0.rank3_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 3. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>3 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
2	rank2_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 2. Writing to this bit causes OPREFSTAT0.rank2_refresh_busy to be set. When OPREFSTAT0.rank2_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 2. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>2 **Testable:** readOnly **Programming Mode:** Dynamic
1	rank1_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 1. Writing to this bit causes OPREFSTAT0.rank1_refresh_busy to be set. When OPREFSTAT0.rank1_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 1. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>1 **Testable:** readOnly **Programming Mode:** Dynamic
0	rank0_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 0. Writing to this bit causes OPREFSTAT0.rank0_refresh_busy to be set. When OPREFSTAT0.rank0_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 0. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>0 **Testable:** readOnly **Programming Mode:** Dynamic

1.2.107 OPREFCTRL1

- **Description:** Refresh Operation Control Register 1
- **Size:** 32 bits
- **Offset:** 0x10b9c
- **Exists:** UMCTL2_NUM_LRANKS_TOTAL>32

This register is in block REGB_DDRC_CH0.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
rank63_refresh	rank62_refresh	rank61_refresh	rank60_refresh	rank59_refresh	rank58_refresh	rank57_refresh	rank56_refresh	rank55_refresh	rank54_refresh	rank53_refresh	rank52_refresh	rank51_refresh	rank50_refresh	rank49_refresh	rank48_refresh	rank47_refresh	rank46_refresh	rank45_refresh	rank44_refresh	rank43_refresh	rank42_refresh	rank41_refresh	rank40_refresh	rank39_refresh	rank38_refresh	rank37_refresh	rank36_refresh	rank35_refresh	rank34_refresh	rank33_refresh	rank32_refresh

Table 1-166 Fields for Register: OPREFCTRL1

Bits	Name	Memory Access	Description
31	rank63_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 63. Writing to this bit causes OPREFSTAT1.rank63_refresh_busy to be set. When OPREFSTAT1.rank63_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 63. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>63 **Testable:** readOnly **Programming Mode:** Dynamic
30	rank62_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 62. Writing to this bit causes OPREFSTAT1.rank62_refresh_busy to be set. When OPREFSTAT1.rank62_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 62. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>62 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
29	rank61_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 61. Writing to this bit causes OPREFSTAT1.rank61_refresh_busy to be set. When OPREFSTAT1.rank61_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 61. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>61 **Testable:** readOnly **Programming Mode:** Dynamic
28	rank60_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 60. Writing to this bit causes OPREFSTAT1.rank60_refresh_busy to be set. When OPREFSTAT1.rank60_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 60. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>60 **Testable:** readOnly **Programming Mode:** Dynamic
27	rank59_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 59. Writing to this bit causes OPREFSTAT1.rank59_refresh_busy to be set. When OPREFSTAT1.rank59_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 59. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>59 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
26	rank58_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 58. Writing to this bit causes OPREFSTAT1.rank58_refresh_busy to be set. When OPREFSTAT1.rank58_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 58. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>58 **Testable:** readOnly **Programming Mode:** Dynamic
25	rank57_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 57. Writing to this bit causes OPREFSTAT1.rank57_refresh_busy to be set. When OPREFSTAT1.rank57_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 57. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>57 **Testable:** readOnly **Programming Mode:** Dynamic
24	rank56_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 56. Writing to this bit causes OPREFSTAT1.rank56_refresh_busy to be set. When OPREFSTAT1.rank56_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 56. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>56 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
23	rank55_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 55. Writing to this bit causes OPREFSTAT1.rank55_refresh_busy to be set. When OPREFSTAT1.rank55_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 55. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>55 **Testable:** readOnly **Programming Mode:** Dynamic
22	rank54_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 54. Writing to this bit causes OPREFSTAT1.rank54_refresh_busy to be set. When OPREFSTAT1.rank54_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 54. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>54 **Testable:** readOnly **Programming Mode:** Dynamic
21	rank53_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 53. Writing to this bit causes OPREFSTAT1.rank53_refresh_busy to be set. When OPREFSTAT1.rank53_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 53. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>53 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
20	rank52_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 52. Writing to this bit causes OPREFSTAT1.rank52_refresh_busy to be set. When OPREFSTAT1.rank52_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 52. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>52 **Testable:** readOnly **Programming Mode:** Dynamic
19	rank51_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 51. Writing to this bit causes OPREFSTAT1.rank51_refresh_busy to be set. When OPREFSTAT1.rank51_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 51. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>51 **Testable:** readOnly **Programming Mode:** Dynamic
18	rank50_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 50. Writing to this bit causes OPREFSTAT1.rank50_refresh_busy to be set. When OPREFSTAT1.rank50_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 50. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>50 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
17	rank49_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 49. Writing to this bit causes OPREFSTAT1.rank49_refresh_busy to be set. When OPREFSTAT1.rank49_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 49. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>49 **Testable:** readOnly **Programming Mode:** Dynamic
16	rank48_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 48. Writing to this bit causes OPREFSTAT1.rank48_refresh_busy to be set. When OPREFSTAT1.rank48_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 48. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>48 **Testable:** readOnly **Programming Mode:** Dynamic
15	rank47_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 47. Writing to this bit causes OPREFSTAT1.rank47_refresh_busy to be set. When OPREFSTAT1.rank47_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 47. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>47 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
14	rank46_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 46. Writing to this bit causes OPREFSTAT1.rank46_refresh_busy to be set. When OPREFSTAT1.rank46_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 46. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>46 **Testable:** readOnly **Programming Mode:** Dynamic
13	rank45_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 45. Writing to this bit causes OPREFSTAT1.rank45_refresh_busy to be set. When OPREFSTAT1.rank45_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 45. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>45 **Testable:** readOnly **Programming Mode:** Dynamic
12	rank44_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 44. Writing to this bit causes OPREFSTAT1.rank44_refresh_busy to be set. When OPREFSTAT1.rank44_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 44. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>44 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
11	rank43_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 43. Writing to this bit causes OPREFSTAT1.rank43_refresh_busy to be set. When OPREFSTAT1.rank43_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 43. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>43 **Testable:** readOnly **Programming Mode:** Dynamic
10	rank42_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 42. Writing to this bit causes OPREFSTAT1.rank42_refresh_busy to be set. When OPREFSTAT1.rank42_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 42. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>42 **Testable:** readOnly **Programming Mode:** Dynamic
9	rank41_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 41. Writing to this bit causes OPREFSTAT1.rank41_refresh_busy to be set. When OPREFSTAT1.rank41_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 41. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>41 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
8	rank40_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 40. Writing to this bit causes OPREFSTAT1.rank40_refresh_busy to be set. When OPREFSTAT1.rank40_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 40. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>40 **Testable:** readOnly **Programming Mode:** Dynamic
7	rank39_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 39. Writing to this bit causes OPREFSTAT1.rank39_refresh_busy to be set. When OPREFSTAT1.rank39_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 39. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>39 **Testable:** readOnly **Programming Mode:** Dynamic
6	rank38_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 38. Writing to this bit causes OPREFSTAT1.rank38_refresh_busy to be set. When OPREFSTAT1.rank38_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 38. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>38 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
5	rank37_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 37. Writing to this bit causes OPREFSTAT1.rank37_refresh_busy to be set. When OPREFSTAT1.rank37_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 37. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>37 **Testable:** readOnly **Programming Mode:** Dynamic
4	rank36_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 36. Writing to this bit causes OPREFSTAT1.rank36_refresh_busy to be set. When OPREFSTAT1.rank36_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 36. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>36 **Testable:** readOnly **Programming Mode:** Dynamic
3	rank35_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 35. Writing to this bit causes OPREFSTAT1.rank35_refresh_busy to be set. When OPREFSTAT1.rank35_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 35. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>35 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
2	rank34_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 34. Writing to this bit causes OPREFSTAT1.rank34_refresh_busy to be set. When OPREFSTAT1.rank34_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 34. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>34 **Testable:** readOnly **Programming Mode:** Dynamic
1	rank33_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 33. Writing to this bit causes OPREFSTAT1.rank33_refresh_busy to be set. When OPREFSTAT1.rank33_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 33. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>33 **Testable:** readOnly **Programming Mode:** Dynamic
0	rank32_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 32. Writing to this bit causes OPREFSTAT1.rank32_refresh_busy to be set. When OPREFSTAT1.rank32_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 32. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>32 **Testable:** readOnly **Programming Mode:** Dynamic

1.2.108 OPREFSTAT0

- **Description:** Refresh Operation Status Register 0
- **Size:** 32 bits
- **Offset:** 0x10ba0
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
rank31_refresh_busy	rank30_refresh_busy	rank29_refresh_busy	rank28_refresh_busy	rank27_refresh_busy	rank26_refresh_busy	rank25_refresh_busy	rank24_refresh_busy	rank23_refresh_busy	rank22_refresh_busy	rank21_refresh_busy	rank20_refresh_busy	rank19_refresh_busy	rank18_refresh_busy	rank17_refresh_busy	rank16_refresh_busy	rank15_refresh_busy	rank14_refresh_busy	rank13_refresh_busy	rank12_refresh_busy	rank11_refresh_busy	rank10_refresh_busy	rank9_refresh_busy	rank8_refresh_busy	rank7_refresh_busy	rank6_refresh_busy	rank5_refresh_busy	rank4_refresh_busy	rank3_refresh_busy	rank2_refresh_busy	rank1_refresh_busy	rank0_refresh_busy

Table 1-167 Fields for Register: OPREFSTAT0

Bits	Name	Memory Access	Description
31	rank31_refresh_busy	R	SoC core may initiate a rank31_refresh operation (refresh operation to rank 31) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank31_refresh is set to one. It goes low when the rank31_refresh operation is stored in the DDRCTL. It is recommended not to perform rank31_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank31_refresh operation ■ 1 - Indicates that rank31_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>31 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
30	rank30_refresh_busy	R	SoC core may initiate a rank30_refresh operation (refresh operation to rank 30) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank30_refresh is set to one. It goes low when the rank30_refresh operation is stored in the DDRCTL. It is recommended not to perform rank30_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank30_refresh operation ■ 1 - Indicates that rank30_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>30 **Programming Mode:** Dynamic
29	rank29_refresh_busy	R	SoC core may initiate a rank29_refresh operation (refresh operation to rank 29) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank29_refresh is set to one. It goes low when the rank29_refresh operation is stored in the DDRCTL. It is recommended not to perform rank29_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank29_refresh operation ■ 1 - Indicates that rank29_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>29 **Programming Mode:** Dynamic
28	rank28_refresh_busy	R	SoC core may initiate a rank28_refresh operation (refresh operation to rank 28) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank28_refresh is set to one. It goes low when the rank28_refresh operation is stored in the DDRCTL. It is recommended not to perform rank28_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank28_refresh operation ■ 1 - Indicates that rank28_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>28 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
27	rank27_refresh_busy	R	SoC core may initiate a rank27_refresh operation (refresh operation to rank 27) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank27_refresh is set to one. It goes low when the rank27_refresh operation is stored in the DDRCTL. It is recommended not to perform rank27_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank27_refresh operation ■ 1 - Indicates that rank27_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>27 **Programming Mode:** Dynamic
26	rank26_refresh_busy	R	SoC core may initiate a rank26_refresh operation (refresh operation to rank 26) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank26_refresh is set to one. It goes low when the rank26_refresh operation is stored in the DDRCTL. It is recommended not to perform rank26_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank26_refresh operation ■ 1 - Indicates that rank26_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>26 **Programming Mode:** Dynamic
25	rank25_refresh_busy	R	SoC core may initiate a rank25_refresh operation (refresh operation to rank 25) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank25_refresh is set to one. It goes low when the rank25_refresh operation is stored in the DDRCTL. It is recommended not to perform rank25_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank25_refresh operation ■ 1 - Indicates that rank25_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>25 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
24	rank24_refresh_busy	R	SoC core may initiate a rank24_refresh operation (refresh operation to rank 24) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank24_refresh is set to one. It goes low when the rank24_refresh operation is stored in the DDRCTL. It is recommended not to perform rank24_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank24_refresh operation ■ 1 - Indicates that rank24_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>24 **Programming Mode:** Dynamic
23	rank23_refresh_busy	R	SoC core may initiate a rank23_refresh operation (refresh operation to rank 23) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank23_refresh is set to one. It goes low when the rank23_refresh operation is stored in the DDRCTL. It is recommended not to perform rank23_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank23_refresh operation ■ 1 - Indicates that rank23_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>23 **Programming Mode:** Dynamic
22	rank22_refresh_busy	R	SoC core may initiate a rank22_refresh operation (refresh operation to rank 22) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank22_refresh is set to one. It goes low when the rank22_refresh operation is stored in the DDRCTL. It is recommended not to perform rank22_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank22_refresh operation ■ 1 - Indicates that rank22_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>22 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
21	rank21_refresh_busy	R	SoC core may initiate a rank21_refresh operation (refresh operation to rank 21) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank21_refresh is set to one. It goes low when the rank21_refresh operation is stored in the DDRCTL. It is recommended not to perform rank21_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank21_refresh operation ■ 1 - Indicates that rank21_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>21 **Programming Mode:** Dynamic
20	rank20_refresh_busy	R	SoC core may initiate a rank20_refresh operation (refresh operation to rank 20) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank20_refresh is set to one. It goes low when the rank20_refresh operation is stored in the DDRCTL. It is recommended not to perform rank20_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank20_refresh operation ■ 1 - Indicates that rank20_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>20 **Programming Mode:** Dynamic
19	rank19_refresh_busy	R	SoC core may initiate a rank19_refresh operation (refresh operation to rank 19) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank19_refresh is set to one. It goes low when the rank19_refresh operation is stored in the DDRCTL. It is recommended not to perform rank19_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank19_refresh operation ■ 1 - Indicates that rank19_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>19 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
18	rank18_refresh_busy	R	SoC core may initiate a rank18_refresh operation (refresh operation to rank 18) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank18_refresh is set to one. It goes low when the rank18_refresh operation is stored in the DDRCTL. It is recommended not to perform rank18_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank18_refresh operation ■ 1 - Indicates that rank18_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>18 **Programming Mode:** Dynamic
17	rank17_refresh_busy	R	SoC core may initiate a rank17_refresh operation (refresh operation to rank 17) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank17_refresh is set to one. It goes low when the rank17_refresh operation is stored in the DDRCTL. It is recommended not to perform rank17_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank17_refresh operation ■ 1 - Indicates that rank17_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>17 **Programming Mode:** Dynamic
16	rank16_refresh_busy	R	SoC core may initiate a rank16_refresh operation (refresh operation to rank 16) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank16_refresh is set to one. It goes low when the rank16_refresh operation is stored in the DDRCTL. It is recommended not to perform rank16_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank16_refresh operation ■ 1 - Indicates that rank16_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>16 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
15	rank15_refresh_busy	R	SoC core may initiate a rank15_refresh operation (refresh operation to rank 15) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank15_refresh is set to one. It goes low when the rank15_refresh operation is stored in the DDRCTL. It is recommended not to perform rank15_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank15_refresh operation ■ 1 - Indicates that rank15_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>15 **Programming Mode:** Dynamic
14	rank14_refresh_busy	R	SoC core may initiate a rank14_refresh operation (refresh operation to rank 14) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank14_refresh is set to one. It goes low when the rank14_refresh operation is stored in the DDRCTL. It is recommended not to perform rank14_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank14_refresh operation ■ 1 - Indicates that rank14_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>14 **Programming Mode:** Dynamic
13	rank13_refresh_busy	R	SoC core may initiate a rank13_refresh operation (refresh operation to rank 13) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank13_refresh is set to one. It goes low when the rank13_refresh operation is stored in the DDRCTL. It is recommended not to perform rank13_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank13_refresh operation ■ 1 - Indicates that rank13_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>13 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
12	rank12_refresh_busy	R	SoC core may initiate a rank12_refresh operation (refresh operation to rank 12) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank12_refresh is set to one. It goes low when the rank12_refresh operation is stored in the DDRCTL. It is recommended not to perform rank12_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank12_refresh operation ■ 1 - Indicates that rank12_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>12 **Programming Mode:** Dynamic
11	rank11_refresh_busy	R	SoC core may initiate a rank11_refresh operation (refresh operation to rank 11) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank11_refresh is set to one. It goes low when the rank11_refresh operation is stored in the DDRCTL. It is recommended not to perform rank11_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank11_refresh operation ■ 1 - Indicates that rank11_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>11 **Programming Mode:** Dynamic
10	rank10_refresh_busy	R	SoC core may initiate a rank10_refresh operation (refresh operation to rank 10) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank10_refresh is set to one. It goes low when the rank10_refresh operation is stored in the DDRCTL. It is recommended not to perform rank10_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank10_refresh operation ■ 1 - Indicates that rank10_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>10 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
9	rank9_refresh_busy	R	SoC core may initiate a rank9_refresh operation (refresh operation to rank 9) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank9_refresh is set to one. It goes low when the rank9_refresh operation is stored in the DDRCTL. It is recommended not to perform rank9_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank9_refresh operation ■ 1 - Indicates that rank9_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>9 **Programming Mode:** Dynamic
8	rank8_refresh_busy	R	SoC core may initiate a rank8_refresh operation (refresh operation to rank 8) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank8_refresh is set to one. It goes low when the rank8_refresh operation is stored in the DDRCTL. It is recommended not to perform rank8_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank8_refresh operation ■ 1 - Indicates that rank8_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>8 **Programming Mode:** Dynamic
7	rank7_refresh_busy	R	SoC core may initiate a rank7_refresh operation (refresh operation to rank 7) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank7_refresh is set to one. It goes low when the rank7_refresh operation is stored in the DDRCTL. It is recommended not to perform rank7_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank7_refresh operation ■ 1 - Indicates that rank7_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>7 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
6	rank6_refresh_busy	R	SoC core may initiate a rank6_refresh operation (refresh operation to rank 6) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank6_refresh is set to one. It goes low when the rank6_refresh operation is stored in the DDRCTL. It is recommended not to perform rank6_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank6_refresh operation ■ 1 - Indicates that rank6_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>6 **Programming Mode:** Dynamic
5	rank5_refresh_busy	R	SoC core may initiate a rank5_refresh operation (refresh operation to rank 5) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank5_refresh is set to one. It goes low when the rank5_refresh operation is stored in the DDRCTL. It is recommended not to perform rank5_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank5_refresh operation ■ 1 - Indicates that rank5_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>5 **Programming Mode:** Dynamic
4	rank4_refresh_busy	R	SoC core may initiate a rank4_refresh operation (refresh operation to rank 4) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank4_refresh is set to one. It goes low when the rank4_refresh operation is stored in the DDRCTL. It is recommended not to perform rank4_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank4_refresh operation ■ 1 - Indicates that rank4_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>4 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
3	rank3_refresh_busy	R	SoC core may initiate a rank3_refresh operation (refresh operation to rank 3) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank3_refresh is set to one. It goes low when the rank3_refresh operation is stored in the DDRCTL. It is recommended not to perform rank3_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank3_refresh operation ■ 1 - Indicates that rank3_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>3 **Programming Mode:** Dynamic
2	rank2_refresh_busy	R	SoC core may initiate a rank2_refresh operation (refresh operation to rank 2) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank2_refresh is set to one. It goes low when the rank2_refresh operation is stored in the DDRCTL. It is recommended not to perform rank2_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank2_refresh operation ■ 1 - Indicates that rank2_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>2 **Programming Mode:** Dynamic
1	rank1_refresh_busy	R	SoC core may initiate a rank1_refresh operation (refresh operation to rank 1) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank1_refresh is set to one. It goes low when the rank1_refresh operation is stored in the DDRCTL. It is recommended not to perform rank1_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank1_refresh operation ■ 1 - Indicates that rank1_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
0	rank0_refresh_busy	R	SoC core may initiate a rank0_refresh operation (refresh operation to rank 0) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank0_refresh is set to one. It goes low when the rank0_refresh operation is stored in the DDRCTL. It is recommended not to perform rank0_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank0_refresh operation ■ 1 - Indicates that rank0_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>0 **Programming Mode:** Dynamic

1.2.109 OPREFSTAT1

- **Description:** Refresh Operation Status Register 1
- **Size:** 32 bits
- **Offset:** 0x10ba4
- **Exists:** UMCTL2_NUM_LRANKS_TOTAL>32

This register is in block REGB_DDRC_CH0.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
rank63_refresh_busy	rank62_refresh_busy	rank61_refresh_busy	rank60_refresh_busy	rank59_refresh_busy	rank58_refresh_busy	rank57_refresh_busy	rank56_refresh_busy	rank55_refresh_busy	rank54_refresh_busy	rank53_refresh_busy	rank52_refresh_busy	rank51_refresh_busy	rank50_refresh_busy	rank49_refresh_busy	rank48_refresh_busy	rank47_refresh_busy	rank46_refresh_busy	rank45_refresh_busy	rank44_refresh_busy	rank43_refresh_busy	rank42_refresh_busy	rank41_refresh_busy	rank40_refresh_busy	rank39_refresh_busy	rank38_refresh_busy	rank37_refresh_busy	rank36_refresh_busy	rank35_refresh_busy	rank34_refresh_busy	rank33_refresh_busy	rank32_refresh_busy

Table 1-168 Fields for Register: OPREFSTAT1

Bits	Name	Memory Access	Description
31	rank63_refresh_busy	R	SoC core may initiate a rank63_refresh operation (refresh operation to rank 63) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank63_refresh is set to one. It goes low when the rank63_refresh operation is stored in the DDRCTL. It is recommended not to perform rank63_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank63_refresh operation ■ 1 - Indicates that rank63_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>63 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
30	rank62_refresh_busy	R	SoC core may initiate a rank62_refresh operation (refresh operation to rank 62) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank62_refresh is set to one. It goes low when the rank62_refresh operation is stored in the DDRCTL. It is recommended not to perform rank62_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank62_refresh operation ■ 1 - Indicates that rank62_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>62 **Programming Mode:** Dynamic
29	rank61_refresh_busy	R	SoC core may initiate a rank61_refresh operation (refresh operation to rank 61) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank61_refresh is set to one. It goes low when the rank61_refresh operation is stored in the DDRCTL. It is recommended not to perform rank61_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank61_refresh operation ■ 1 - Indicates that rank61_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>61 **Programming Mode:** Dynamic
28	rank60_refresh_busy	R	SoC core may initiate a rank60_refresh operation (refresh operation to rank 60) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank60_refresh is set to one. It goes low when the rank60_refresh operation is stored in the DDRCTL. It is recommended not to perform rank60_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank60_refresh operation ■ 1 - Indicates that rank60_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>60 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
27	rank59_refresh_busy	R	SoC core may initiate a rank59_refresh operation (refresh operation to rank 59) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank59_refresh is set to one. It goes low when the rank59_refresh operation is stored in the DDRCTL. It is recommended not to perform rank59_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank59_refresh operation ■ 1 - Indicates that rank59_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>59 **Programming Mode:** Dynamic
26	rank58_refresh_busy	R	SoC core may initiate a rank58_refresh operation (refresh operation to rank 58) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank58_refresh is set to one. It goes low when the rank58_refresh operation is stored in the DDRCTL. It is recommended not to perform rank58_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank58_refresh operation ■ 1 - Indicates that rank58_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>58 **Programming Mode:** Dynamic
25	rank57_refresh_busy	R	SoC core may initiate a rank57_refresh operation (refresh operation to rank 57) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank57_refresh is set to one. It goes low when the rank57_refresh operation is stored in the DDRCTL. It is recommended not to perform rank57_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank57_refresh operation ■ 1 - Indicates that rank57_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>57 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
24	rank56_refresh_busy	R	SoC core may initiate a rank56_refresh operation (refresh operation to rank 56) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank56_refresh is set to one. It goes low when the rank56_refresh operation is stored in the DDRCTL. It is recommended not to perform rank56_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank56_refresh operation ■ 1 - Indicates that rank56_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>56 **Programming Mode:** Dynamic
23	rank55_refresh_busy	R	SoC core may initiate a rank55_refresh operation (refresh operation to rank 55) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank55_refresh is set to one. It goes low when the rank55_refresh operation is stored in the DDRCTL. It is recommended not to perform rank55_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank55_refresh operation ■ 1 - Indicates that rank55_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>55 **Programming Mode:** Dynamic
22	rank54_refresh_busy	R	SoC core may initiate a rank54_refresh operation (refresh operation to rank 54) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank54_refresh is set to one. It goes low when the rank54_refresh operation is stored in the DDRCTL. It is recommended not to perform rank54_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank54_refresh operation ■ 1 - Indicates that rank54_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>54 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
21	rank53_refresh_busy	R	SoC core may initiate a rank53_refresh operation (refresh operation to rank 53) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank53_refresh is set to one. It goes low when the rank53_refresh operation is stored in the DDRCTL. It is recommended not to perform rank53_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank53_refresh operation ■ 1 - Indicates that rank53_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>53 **Programming Mode:** Dynamic
20	rank52_refresh_busy	R	SoC core may initiate a rank52_refresh operation (refresh operation to rank 52) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank52_refresh is set to one. It goes low when the rank52_refresh operation is stored in the DDRCTL. It is recommended not to perform rank52_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank52_refresh operation ■ 1 - Indicates that rank52_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>52 **Programming Mode:** Dynamic
19	rank51_refresh_busy	R	SoC core may initiate a rank51_refresh operation (refresh operation to rank 51) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank51_refresh is set to one. It goes low when the rank51_refresh operation is stored in the DDRCTL. It is recommended not to perform rank51_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank51_refresh operation ■ 1 - Indicates that rank51_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>51 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
18	rank50_refresh_busy	R	SoC core may initiate a rank50_refresh operation (refresh operation to rank 50) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank50_refresh is set to one. It goes low when the rank50_refresh operation is stored in the DDRCTL. It is recommended not to perform rank50_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank50_refresh operation ■ 1 - Indicates that rank50_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>50 **Programming Mode:** Dynamic
17	rank49_refresh_busy	R	SoC core may initiate a rank49_refresh operation (refresh operation to rank 49) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank49_refresh is set to one. It goes low when the rank49_refresh operation is stored in the DDRCTL. It is recommended not to perform rank49_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank49_refresh operation ■ 1 - Indicates that rank49_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>49 **Programming Mode:** Dynamic
16	rank48_refresh_busy	R	SoC core may initiate a rank48_refresh operation (refresh operation to rank 48) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank48_refresh is set to one. It goes low when the rank48_refresh operation is stored in the DDRCTL. It is recommended not to perform rank48_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank48_refresh operation ■ 1 - Indicates that rank48_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>48 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
15	rank47_refresh_busy	R	SoC core may initiate a rank47_refresh operation (refresh operation to rank 47) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank47_refresh is set to one. It goes low when the rank47_refresh operation is stored in the DDRCTL. It is recommended not to perform rank47_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank47_refresh operation ■ 1 - Indicates that rank47_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>47 **Programming Mode:** Dynamic
14	rank46_refresh_busy	R	SoC core may initiate a rank46_refresh operation (refresh operation to rank 46) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank46_refresh is set to one. It goes low when the rank46_refresh operation is stored in the DDRCTL. It is recommended not to perform rank46_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank46_refresh operation ■ 1 - Indicates that rank46_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>46 **Programming Mode:** Dynamic
13	rank45_refresh_busy	R	SoC core may initiate a rank45_refresh operation (refresh operation to rank 45) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank45_refresh is set to one. It goes low when the rank45_refresh operation is stored in the DDRCTL. It is recommended not to perform rank45_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank45_refresh operation ■ 1 - Indicates that rank45_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>45 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
12	rank44_refresh_busy	R	SoC core may initiate a rank44_refresh operation (refresh operation to rank 44) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank44_refresh is set to one. It goes low when the rank44_refresh operation is stored in the DDRCTL. It is recommended not to perform rank44_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank44_refresh operation ■ 1 - Indicates that rank44_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>44 **Programming Mode:** Dynamic
11	rank43_refresh_busy	R	SoC core may initiate a rank43_refresh operation (refresh operation to rank 43) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank43_refresh is set to one. It goes low when the rank43_refresh operation is stored in the DDRCTL. It is recommended not to perform rank43_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank43_refresh operation ■ 1 - Indicates that rank43_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>43 **Programming Mode:** Dynamic
10	rank42_refresh_busy	R	SoC core may initiate a rank42_refresh operation (refresh operation to rank 42) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank42_refresh is set to one. It goes low when the rank42_refresh operation is stored in the DDRCTL. It is recommended not to perform rank42_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank42_refresh operation ■ 1 - Indicates that rank42_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>42 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
9	rank41_refresh_busy	R	SoC core may initiate a rank41_refresh operation (refresh operation to rank 41) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank41_refresh is set to one. It goes low when the rank41_refresh operation is stored in the DDRCTL. It is recommended not to perform rank41_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank41_refresh operation ■ 1 - Indicates that rank41_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>41 **Programming Mode:** Dynamic
8	rank40_refresh_busy	R	SoC core may initiate a rank40_refresh operation (refresh operation to rank 40) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank40_refresh is set to one. It goes low when the rank40_refresh operation is stored in the DDRCTL. It is recommended not to perform rank40_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank40_refresh operation ■ 1 - Indicates that rank40_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>40 **Programming Mode:** Dynamic
7	rank39_refresh_busy	R	SoC core may initiate a rank39_refresh operation (refresh operation to rank 39) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank39_refresh is set to one. It goes low when the rank39_refresh operation is stored in the DDRCTL. It is recommended not to perform rank39_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank39_refresh operation ■ 1 - Indicates that rank39_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>39 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
6	rank38_refresh_busy	R	SoC core may initiate a rank38_refresh operation (refresh operation to rank 38) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank38_refresh is set to one. It goes low when the rank38_refresh operation is stored in the DDRCTL. It is recommended not to perform rank38_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank38_refresh operation ■ 1 - Indicates that rank38_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>38 **Programming Mode:** Dynamic
5	rank37_refresh_busy	R	SoC core may initiate a rank37_refresh operation (refresh operation to rank 37) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank37_refresh is set to one. It goes low when the rank37_refresh operation is stored in the DDRCTL. It is recommended not to perform rank37_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank37_refresh operation ■ 1 - Indicates that rank37_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>37 **Programming Mode:** Dynamic
4	rank36_refresh_busy	R	SoC core may initiate a rank36_refresh operation (refresh operation to rank 36) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank36_refresh is set to one. It goes low when the rank36_refresh operation is stored in the DDRCTL. It is recommended not to perform rank36_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank36_refresh operation ■ 1 - Indicates that rank36_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>36 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
3	rank35_refresh_busy	R	SoC core may initiate a rank35_refresh operation (refresh operation to rank 35) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank35_refresh is set to one. It goes low when the rank35_refresh operation is stored in the DDRCTL. It is recommended not to perform rank35_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank35_refresh operation ■ 1 - Indicates that rank35_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>35 **Programming Mode:** Dynamic
2	rank34_refresh_busy	R	SoC core may initiate a rank34_refresh operation (refresh operation to rank 34) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank34_refresh is set to one. It goes low when the rank34_refresh operation is stored in the DDRCTL. It is recommended not to perform rank34_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank34_refresh operation ■ 1 - Indicates that rank34_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>34 **Programming Mode:** Dynamic
1	rank33_refresh_busy	R	SoC core may initiate a rank33_refresh operation (refresh operation to rank 33) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank33_refresh is set to one. It goes low when the rank33_refresh operation is stored in the DDRCTL. It is recommended not to perform rank33_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank33_refresh operation ■ 1 - Indicates that rank33_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>33 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
0	rank32_refresh_busy	R	SoC core may initiate a rank32_refresh operation (refresh operation to rank 32) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank32_refresh is set to one. It goes low when the rank32_refresh operation is stored in the DDRCTL. It is recommended not to perform rank32_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank32_refresh operation ■ 1 - Indicates that rank32_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>32 **Programming Mode:** Dynamic

1.2.110 SWCTL

- **Description:** Software Register Programming Control Enable
- **Size:** 32 bits
- **Offset:** 0x10c80
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

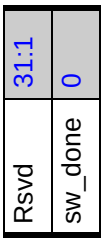


Table 1-169 Fields for Register: SWCTL

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	sw_done	R/W	Enable quasi-dynamic register programming outside reset. Program register to 0 to enable quasi-dynamic programming. Set back register to 1 once programming is done. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Dynamic

1.2.111 SWSTAT

- **Description:** Software Register Programming Control Status
- **Size:** 32 bits
- **Offset:** 0x10c84
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

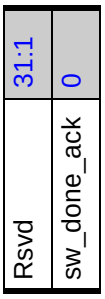


Table 1-170 Fields for Register: SWSTAT

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	sw_done_ack	R	Register programming done. This register is the echo of SWCTL.sw_done. Wait for sw_done value 1 to propagate to sw_done_ack at the end of the programming sequence to ensure that the correct registers values are propagated to the destination clock domains. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Static

1.2.112 RANKCTL

- **Description:** Rank Control Register
- **Size:** 32 bits
- **Offset:** 0x10c90
- **Exists:** UMCTL2_NUM_LRANKS_TOTAL>1

This register is in block REGB_DDRC_CH0.

31:24	23:20	19:16	15:12	11:4	3:0
Rsvd	max_logical_rank_wr	max_logical_rank_rd	max_rank_wr	Rsvd	max_rank_rd

Table 1-171 Fields for Register: RANKCTL

Bits	Name	Memory Access	Description
31:24			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
23:20	max_logical_rank_wr	R/W	Present for DDR4/DDR5 3DS supported configurations. Background: Writes to the same logical rank can be performed back-to-back. Writes to different logical ranks may require additional gap in case DRAMSET1TMG16.t_ccd_dlr is larger than DRAMSET1TMG9.t_ccd_s. The DDRCTL arbitrates for bus access on a cycle-by-cycle basis; therefore after a write is scheduled, there are few clock cycles (determined by DRAMSET1TMG16.t_ccd_dlr - DRAMSET1TMG9.t_ccd_s) in which only writes from the same logical rank(but different bank group) are eligible to be scheduled. This prevents writes from other logical ranks from having fair access to the data bus. This parameter represents the maximum number of writes that can be scheduled consecutively to the same logical rank(but different bank group). After this number is reached, - DDR4: a delay equal to (DRAMSET1TMG16.t_ccd_dlr - DRAMSET1TMG9.t_ccd_s) is inserted by the scheduler to allow all logical ranks a fair opportunity to be scheduled. - DDR5: writes to the same logical rank are blocked until read-write mode turn around or writes are issued to other logical ranks or other ranks. Higher numbers increase bandwidth utilization, lower numbers increase fairness. This feature can be DISABLED by setting this register to 0. When set to 0, the Controller will stay on the same logical rank as long as commands are available for it. Minimum programmable value is 0 (feature disabled) and maximum programmable value is 0xF. This register setting is ignored when MSTR0.active_logical_ranks=0. **Value After Reset:** 0x0 **Exists:** UMCTL2_CID_EN==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
19:16	max_logical_rank_rd	R/W	Present for DDR4/DDR5 3DS supported configurations. Background: Reads to the same logical rank can be performed back-to-back. Reads to different logical ranks may require additional gap in case DRAMSET1TMG16.t_ccd_dlr is larger than DRAMSET1TMG9.t_ccd_s. The DDRCTL arbitrates for bus access on a cycle-by-cycle basis; therefore after a read is scheduled, there are few clock cycles (determined by DRAMSET1TMG16.t_ccd_dlr - DRAMSET1TMG9.t_ccd_s) in which only reads from the same logical rank(but different bank group) are eligible to be scheduled. This prevents reads from other logical ranks from having fair access to the data bus. This parameter represents the maximum number of reads that can be scheduled consecutively to the same logical rank(but different bank group). After this number is reached, - DDR4: a delay equal to (DRAMSET1TMG16.t_ccd_dlr - DRAMSET1TMG9.t_ccd_s) is inserted by the scheduler to allow all logical ranks a fair opportunity to be scheduled. - DDR5: reads to the same logical rank are blocked until read-write mode turn around or reads are issued to other logical ranks or other ranks. Higher numbers increase bandwidth utilization, lower numbers increase fairness. This feature can be DISABLED by setting this register to 0. When set to 0, the Controller will stay on the same logical rank as long as commands are available for it. Minimum programmable value is 0 (feature disabled) and maximum programmable value is 0xF. This register setting is ignored when MSTR0.active_logical_ranks=0. **Value After Reset:** 0xf **Exists:** UMCTL2_CID_EN==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
15:12	max_rank_wr	R/W	Only present for multi-rank configurations. Background: Writes to the same rank can be performed back-to-back. Writes to different ranks require additional gap dictated by the register RANKCTL.diff_rank_wr_gap. This is to avoid possible data bus contention as well as to give PHY enough time to switch the delay when changing ranks. The DDRCTL arbitrates for bus access on a cycle-by-cycle basis; therefore after a write is scheduled, there are few clock cycles (determined by the value on RANKCTL.diff_rank_wr_gap register) in which only writes from the same rank are eligible to be scheduled. This prevents writes from other ranks from having fair access to the data bus. This parameter represents the maximum number of writes that can be scheduled consecutively to the same rank. After this number is reached, - DDR4/LPDDR: a delay equal to RANKCTL.diff_rank_wr_gap is inserted by the scheduler to allow all ranks a fair opportunity to be scheduled. Higher numbers increase bandwidth utilization, lower numbers increase fairness. - DDR5: writes to the same rank are blocked until read-write mode turn around or writes are issued to other ranks. This feature can be DISABLED by setting this register to 0. When set to 0, the Controller will stay on the same rank as long as commands are available for it. Minimum programmable value is 0 (feature disabled) and maximum programmable value is 0xF. FOR PERFORMANCE ONLY. **Value After Reset:** 0x0 **Exists:** MEMC_NUM_RANKS>1 **Programming Mode:** Static
11:4			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
3:0	max_rank_rd	R/W	Only present for multi-rank configurations. Background: Reads to the same rank can be performed back-to-back. Reads to different ranks require additional gap dictated by the register RANKCTL.diff_rank_rd_gap. This is to avoid possible data bus contention as well as to give PHY enough time to switch the delay when changing ranks. The DDRCTL arbitrates for bus access on a cycle-by-cycle basis; therefore after a read is scheduled, there are few clock cycles (determined by the value on RANKCTL.diff_rank_rd_gap register) in which only reads from the same rank are eligible to be scheduled. This prevents reads from other ranks from having fair access to the data bus. This parameter represents the maximum number of reads that can be scheduled consecutively to the same rank. After this number is reached, - DDR4/LPDDR: a delay equal to RANKCTL.diff_rank_rd_gap is inserted by the scheduler to allow all ranks a fair opportunity to be scheduled. Higher numbers increase bandwidth utilization, lower numbers increase fairness. - DDR5: reads to the same rank are blocked until read-write mode turn around or reads are issued to other ranks. This feature can be DISABLED by setting this register to 0. When set to 0, the Controller will stay on the same rank as long as commands are available for it. Minimum programmable value is 0 (feature disabled) and maximum programmable value is 0xF. FOR PERFORMANCE ONLY. **Value After Reset:** 0xf **Exists:** MEMC_NUM_RANKS>1 **Programming Mode:** Static

1.2.113 DBICTL

- **Description:** DM/DBI Control Register
- **Size:** 32 bits
- **Offset:** 0x10c94
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

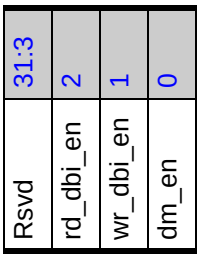


Table 1-172 Fields for Register: DBICTL

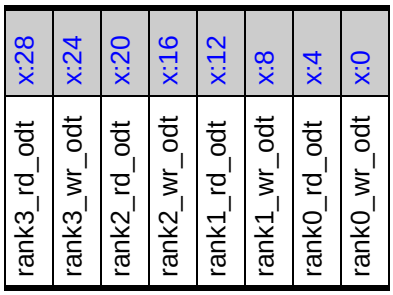
Bits	Name	Memory Access	Description
31:3			**Reserved Field:** Yes
2	rd_dbi_en	R/W	Read DBI enable signal in DDRC. ■ 0 - Read DBI is disabled. ■ 1 - Read DBI is enabled. This signal must be set the same value as DRAM's mode register. ■ DDR4: MR5 bit A12. When x4 devices are used, this signal must be set to 0. ■ DDR5: This signal must be set to 0. ■ LPDDR4/LPDDR5: MR3[6]. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1

Bits	Name	Memory Access	Description
1	wr_dbi_en	R/W	Write DBI enable signal in DDRC. ■ 0 - Write DBI is disabled. ■ 1 - Write DBI is enabled. This signal must be set the same value as DRAM's mode register. ■ DDR4: MR5 bit A11. When x4 devices are used, this signal must be set to 0. ■ DDR5: This signal must be set to 0. ■ LPDDR4/LPDDR5: MR3[7]. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1
0	dm_en	R/W	DM enable signal in DDRC. ■ 0 - DM is disabled. ■ 1 - DM is enabled. This signal must be set the same logical value as DRAM's mode register. ■ DDR4: Set this to same value as MR5 bit A10. When x4 devices are used, this signal must be set to 0. ■ DDR5: Set this to same value as MR5[5]. When x4 devices are used, this signal must be set to 0. ■ LPDDR4/LPDDR5: Set this to inverted value of MR13[5] which is opposite polarity from this signal. **Value After Reset:** 0x1 **Exists:** Always **Volatile:** true **Programming Mode:** Static

1.2.114 ODTMAP

- **Description:** ODT/Rank Map Register
- **Size:** 32 bits
- **Offset:** 0x10c9c
- **Exists:** Always

This register is in block REGB_DDRC_CH0.



x:28	x:24	x:20	x:16	x:12	x:8	x:4	x:0
rank3_rd_odt	rank3_wr_odt	rank2_rd_odt	rank2_wr_odt	rank1_rd_odt	rank1_wr_odt	rank0_rd_odt	rank0_wr_odt

Table 1-173 Fields for Register: ODTMAP

Bits	Name	Memory Access	Description
x:28	rank3_rd_odt	R/W	Indicates which remote ODTs must be turned on during a read from rank 3. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. Present only in configurations that have 4 ranks **Value After Reset:** "(MEMC_NUM_RANKS>=4) ? 0x8 : 0x0" **Exists:** MEMC_NUM_RANKS==4 **Programming Mode:** Static **Range Variable[x]:** "MEMC_NUM_RANKS" + 27
x:24	rank3_wr_odt	R/W	Indicates which remote ODTs must be turned on during a write to rank 3. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. Present only in configurations that have 4 ranks **Value After Reset:** "(MEMC_NUM_RANKS>=4) ? 0x8 : 0x0" **Exists:** MEMC_NUM_RANKS==4 **Programming Mode:** Static **Range Variable[x]:** "MEMC_NUM_RANKS" + 23

Bits	Name	Memory Access	Description
x:20	rank2_rd_odt	R/W	Indicates which remote ODTs must be turned on during a read from rank 2. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. Present only in configurations that have 4 ranks **Value After Reset:** "(MEMC_NUM_RANKS>=4) ? 0x4 : 0x0" **Exists:** MEMC_NUM_RANKS==4 **Programming Mode:** Static **Range Variable[x]:** "MEMC_NUM_RANKS" + 19
x:16	rank2_wr_odt	R/W	Indicates which remote ODTs must be turned on during a write to rank 2. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. Present only in configurations that have 4 ranks **Value After Reset:** "(MEMC_NUM_RANKS>=4) ? 0x4 : 0x0" **Exists:** MEMC_NUM_RANKS==4 **Programming Mode:** Static **Range Variable[x]:** "MEMC_NUM_RANKS" + 15
x:12	rank1_rd_odt	R/W	Indicates which remote ODTs must be turned on during a read from rank 1. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. Present only in configurations that have 2 or more ranks **Value After Reset:** "(MEMC_NUM_RANKS>1) ? 0x2 : 0x0" **Exists:** MEMC_NUM_RANKS>1 **Programming Mode:** Static **Range Variable[x]:** "MEMC_NUM_RANKS" + 11

Bits	Name	Memory Access	Description
x:8	rank1_wr_odt	R/W	Indicates which remote ODTs must be turned on during a write to rank 1. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. Present only in configurations that have 2 or more ranks **Value After Reset:** "(MEMC_NUM_RANKS>1) ? 0x2 : 0x0" **Exists:** MEMC_NUM_RANKS>1 **Programming Mode:** Static **Range Variable[x]:** "MEMC_NUM_RANKS" + 7
x:4	rank0_rd_odt	R/W	Indicates which remote ODTs must be turned on during a read from rank 0. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Static **Range Variable[x]:** "MEMC_NUM_RANKS" + 3
x:0	rank0_wr_odt	R/W	Indicates which remote ODTs must be turned on during a write to rank 0. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Static **Range Variable[x]:** "MEMC_NUM_RANKS" - 1

1.2.115 DATACTL0

- **Description:** Data Control register 0
- **Size:** 32 bits
- **Offset:** 0x10ca0
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_DDRC_CH0.

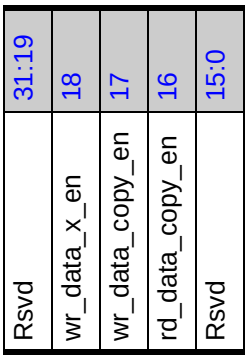


Table 1-174 Fields for Register: DATACTL0

Bits	Name	Memory Access	Description
31:19			**Reserved Field:** Yes
18	wr_data_x_en	R/W	Write Data X 1: Enable Write X 0: Write X This feature is not supported at present. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2
17	wr_data_copy_en	R/W	Write Data Copy 1: Enable Write Data Copy X 0: Disable Write Data Copy X This feature is not supported at present. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2

Bits	Name	Memory Access	Description
16	rd_data_copy_en	R/W	Read Data Copy 1: Enable Read Data Copy X 0: Disable Read Data Copy X This feature is not supported at present. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2
15:0			**Reserved Field:** Yes

1.2.116 SWCTLSTATIC

- **Description:** Static Registers Write Enable
- **Size:** 32 bits
- **Offset:** 0x10ca4
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

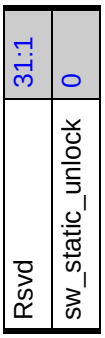


Table 1-175 Fields for Register: SWCTLSTATIC

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	sw_static_unlock	R/W	Enables static register programming outside reset. Program this register to 1 to enable static register programming. Set register back to 0 once programming is done. This register is provided only to be used for software workarounds and it is not meant to be used with all static registers or in all conditions. Unless Synopsys recommends explicitly for a given software sequence, do not use this method to program static registers. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.2.117 INITTMG0

- **Description:** SDRAM Initialization Timing Register 0
- **Size:** 32 bits
- **Offset:** 0x10d00
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

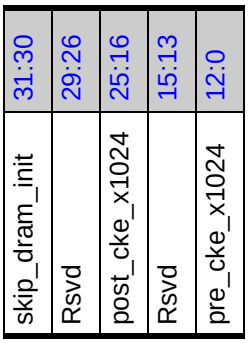


Table 1-176 Fields for Register: INITTMG0

Bits	Name	Memory Access	Description
31:30	skip_dram_init	R/W	If lower bit is enabled the SDRAM initialization routine is skipped. The upper bit decides what state the controller starts up in when reset is removed ■ 00 - SDRAM Initialization routine is run after power-up ■ 01 - SDRAM Initialization routine is skipped after power-up. Controller starts up in Normal Mode ■ 11 - SDRAM Initialization routine is skipped after power-up. Controller starts up in Self-refresh Mode ■ 10 - Reserved. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2
29:26			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
25:16	post_cke_x1024	R/W	Cycles to wait after driving CKE (CS in DDR5) high to start the SDRAM initialization sequence. DDR5: tINIT4 of 2 us (min) - simulation only. LPDDR4: typically requires this to be programmed for a delay of 2 us. LPDDR5: Don't care Unit: Multiples of 1024 DRAM clock cycles. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x2 **Exists:** Always **Programming Mode:** Static
15:13			**Reserved Field:** Yes
12:0	pre_cke_x1024	R/W	Cycles to wait after reset before driving CKE (CS in DDR5) high to start the SDRAM initialization sequence. DDR5: tINIT3 of 4 ms (min) - simulation only. LPDDR4: tINIT3 of 2 ms (min) LPDDR5: Don't care (For LPDDR5, DDRCTL doesn't send the first PDX command to the SDRAM - Assumption is that the first PDX is issued as part of initialization performed by PHY) For DDR4 RDIMMs, this must include the time needed to satisfy tSTAB. Unit: Multiples of 1024 DRAM clock cycles. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x4e **Exists:** Always **Programming Mode:** Static

1.2.118 INITTMG1

- **Description:** SDRAM Initialization Timing Register 1
- **Size:** 32 bits
- **Offset:** 0x10d04
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

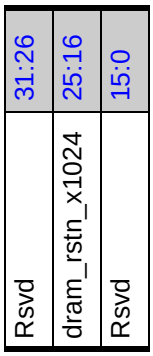


Table 1-177 Fields for Register: INITTMG1

Bits	Name	Memory Access	Description
31:26			**Reserved Field:** Yes
25:16	dram_rstn_x1024	R/W	Number of cycles to assert SDRAM reset signal during init sequence. For use with a Synopsys DDR PHY, this must be set to a minimum of 1. Unit: Multiples of 1024 DRAM clock cycles. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
15:0			**Reserved Field:** Yes

1.2.119 PPT2CTRL0

- **Description:** PPT2 Control Register
- **Size:** 32 bits
- **Offset:** 0x10f00
- **Exists:** DDRCTL_PPT2==1

This register is in block REGB_DDRC_CH0.

31	30:29	28	27:24	23:18	17:12	11:10	9:0
ppt2_wait_ref	Rsvd	ppt2_burst	Rsvd	ppt2_ctrlupd_num_dfi1	ppt2_ctrlupd_num_dfi0	Rsvd	ppt2_burst_num

Table 1-178 Fields for Register: PPT2CTRL0

Bits	Name	Memory Access	Description
31	ppt2_wait_ref	R/W	Wait for REFab/REFpb before sending Normal PPT2. This register must be reset to '1' to bring DDRCTL out of lowest latency with Normal PPT2 enabled. The value can be changed while DFIUPDTMG2.ppt2_en=0 and PPT2STAT0.ppt2_burst_busy=0 ■ 0: Don't wait for REFab/REFpb ■ 1: Always Wait for REFab/REFpb FOR DEBUG ONLY. **Value After Reset:** 0x1 **Exists:** DDRCTL_PPT2==1 **Programming Mode:** Dynamic
30:29			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
28	ppt2_burst	R/W1S	Setting this register bit to 1 triggers a Burst PPT2 operation. It is recommended to set this signal only if in normal operating mode. When the Burst PPT2 operation is complete, the DDRCTL automatically clears this bit. The other register fields of this register must be written in a separate APB transaction, before setting this ppt2_burst bit. **Value After Reset:** 0x0 **Exists:** DDRCTL_PPT2==1 **Testable:** readOnly **Programming Mode:** Dynamic
27:24			**Reserved Field:** Yes
23:18	ppt2_ctrlupd_num_dfi1	R/W	This register indicates the number of times to send ctrlupd_req for DFI1 per retraining_interval in Normal PPT2 operation. The value can be changed while DFIUPDTMG2.ppt2_en=0 and PPT2STAT0.ppt2_burst_busy=0 **Value After Reset:** 0x0 **Exists:** DDRCTL_PPT2==1 **Programming Mode:** Dynamic
17:12	ppt2_ctrlupd_num_dfi0	R/W	This register indicates the number of times to send ctrlupd_req for DFI0 per retraining_interval in Normal PPT2 operation. The value can be changed while DFIUPDTMG2.ppt2_en=0 and PPT2STAT0.ppt2_burst_busy=0 **Value After Reset:** 0x8 **Exists:** DDRCTL_PPT2==1 **Programming Mode:** Dynamic
11:10			**Reserved Field:** Yes
9:0	ppt2_burst_num	R/W	This register indicates the number of times to send ctrlupd_req in Burst PPT2 operation. **Value After Reset:** 0x200 **Exists:** DDRCTL_PPT2==1 **Programming Mode:** Dynamic

1.2.120 PPT2STAT0

- **Description:** Status PPT2 Control Register
- **Size:** 32 bits
- **Offset:** 0x10f10
- **Exists:** DDRCTL_PPT2==1

This register is in block REGB_DDRC_CH0.

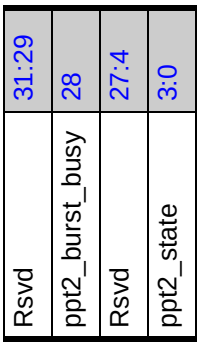


Table 1-179 Fields for Register: PPT2STAT0

Bits	Name	Memory Access	Description
31:29			**Reserved Field:** Yes
28	ppt2_burst_busy	R	SoC core may initiate a Burst PPT2 operation only if this signal is low. This signal goes high in the clock after the DDRCTL accepts the Burst PPT2 request. It goes low when the Burst PPT2 operation is initiated in the DDRCTL. ■ 0 - Indicates that the SoC core can initiate a Burst PPT2 operation ■ 1 - Indicates that Burst PPT2 operation has not been initiated yet or ongoing in the DDRCTL **Value After Reset:** 0x0 **Exists:** DDRCTL_PPT2==1 **Programming Mode:** Dynamic
27:4			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
3:0	ppt2_state	R	This register indicates the state of PPT2 scheduler. ■ 0 - Idle. Neither Normal PPT2 nor Burst PPT2 is enabled/ongoing ■ 1 - Normal PPT2 enabled. Waiting for dfi_t_ctrlupd_interval_type1 expiration ■ 2 - Normal PPT2 ongoing. Waiting for trigger condition; When DDRCTL is in Normal state, waiting for any REF command (according to PPT2CTRL0.ppt2_wait_ref). When DDRCTL is in Automatic SRPD, waiting for a PDX command. ■ 3 - Normal PPT2 ongoing. Waiting for ck stop ready ■ 4 - Normal PPT2 ongoing. Sending PPT2 request to PHY ■ 8 - Burst PPT2 ongoing **Value After Reset:** 0x0 **Exists:** DDRCTL_PPT2==1 **Programming Mode:** Dynamic

1.2.121 DDRCTL_VER_NUMBER

- **Description:** DDRCTL Version Number Register
- **Size:** 32 bits
- **Offset:** 0x10ff8
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

Table 1-180 Fields for Register: DDRCTL_VER_NUMBER

Bits	Name	Memory Access	Description
31:0	ver_number	R	Indicates the Device Version Number value. This is in ASCII format, with each byte corresponding to a character of the version number **Value After Reset:** "DDRCTL_VER_NUMBER_VAL" **Exists:** Always **Programming Mode:** Static

1.2.122 DDRCTL_VER_TYPE

- **Description:** DDRCTL Version Type Register
- **Size:** 32 bits
- **Offset:** 0x10ffc
- **Exists:** Always

This register is in block REGB_DDRC_CH0.

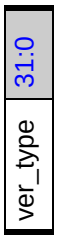


Table 1-181 Fields for Register: DDRCTL_VER_TYPE

Bits	Name	Memory Access	Description
31:0	ver_type	R	Indicates the Device Version Type value. This is in ASCII format, with each byte corresponding to a character of the version type **Value After Reset:** "DDRCTL_VER_TYPE_VAL" **Exists:** Always **Programming Mode:** Static

1.3 REGB_DDRC_CH1 Registers

This register block contains registers related to the control of functionality and the configuration of the DDRC controller. Registers shared by both channels are only present in REGB_DDRC_CH0.

1.3.1 MSTR4

- **Description:** Master Register4
- **Size:** 32 bits
- **Offset:** 0x11010
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_DDRC_CH1.

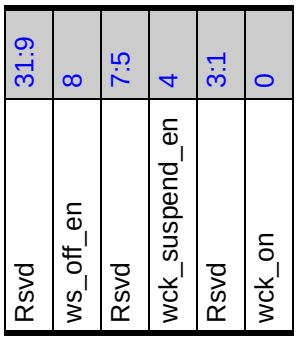


Table 1-182 Fields for Register: MSTR4

Bits	Name	Memory Access	Description
31:9			**Reserved Field:** Yes
8	ws_off_en	R/W	CAS-WS_OFF enable. If this bit is set to 1, the controller actively issues CAS-WS_OFF command. This register is valid only if MSTR4.wck_on is set to 1. **Value After Reset:** 0x1 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2
7:5			**Reserved Field:** Yes
4	wck_suspend_en	R/W	Enhanced WCK always on mode. If this register is set to 1, the controller issues CAS-WCK_SUSPEND command. This register is valid only if MSTR4.wck_on is set to 1. Note: CAS-WCK_SUSPEND command is valid only when MR0 OP[2]=1b(Enhanced WCK Always On mode supported) and MR18 OP[4]=1b(WCK Always On mode enabled). **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2
3:1			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
0	wck_on	R/W	WCK always ON mode ■ 0: WCK Always On mode disabled ■ 1: WCK Always On mode enabled In case of multi-rank system, the controller issues CAS-WS_FS to all ranks to sets DRAM in sync state simultaneously. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2

1.3.2 STAT

- **Description:** Operating Mode Status Register
- **Size:** 32 bits
- **Offset:** 0x11014
- **Exists:** Always

This register is in block REGB_DDRC_CH1.

31:17	16	15	14:12	x:4	3	2:0
Rsvd	selfref_cam_not_empty	Rsvd	selfref_state	selfref_type	Rsvd	operating_mode

Table 1-183 Fields for Register: STAT

Bits	Name	Memory Access	Description
31:17			**Reserved Field:** Yes
16	selfref_cam_not_empty	R	Self refresh with CAMs not empty. Set to 1 when Self Refresh is entered but CAMs are not drained. Cleared after exiting Self Refresh. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
15			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
14:12	selfref_state	R	Self refresh state. This indicates self refresh or self refresh power down state or Deep Sleep Mode (LPDDR5 only). This register is used for frequency change and MRR/MRW access during self refresh. ■ 000 - SDRAM is not in Self Refresh. ■ 001 - Self refresh 1 ■ 010 - Self refresh power down ■ 011 - Self refresh 2 ■ 100 - Deep Sleep Mode (LPDDR5 only) **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Static
x:4	selfref_type	R	Flags if Self Refresh (except LPDDR4/5) or SR-Powerdown (LPDDR4/5) is entered and if it was under Automatic Self Refresh control only or not. ■ 00 - SDRAM is not in Self Refresh (except LPDDR4/5) or SR-Powerdown (LPDDR4/5). If CA parity retry is enabled by RETRYCTL0.capar_retry_enable, this also indicates SRE command is still in parity error window or retry is in-progress. ■ 11 - SDRAM is in Self Refresh (except LPDDR4/5) or SR-Powerdown (LPDDR4/5), which was caused by Automatic Self Refresh only. If retry is enabled, this guarantees SRE command is executed correctly without parity error. ■ 10 - SDRAM is in Self Refresh (except LPDDR4/5) or SR-Powerdown (LPDDR4/5), which was not caused solely under Automatic Self Refresh control. It could have been caused by Hardware Low Power Interface and/or Software (PWRCTL.selfref_sw). If retry is enabled, this guarantees SRE command is executed correctly without parity error. ■ 01 - SDRAM is in Self Refresh, which was caused by PHY Master Request or Normal PPT2. For LPDDR54 and DDR4, only bit[5:4] are used. For DDR5, self-refresh per rank control is supported. ■ bit[5:4] - rank 0 selfref_type ■ bit[7:6] - rank 1 selfref_type ■ bit[9:8] - rank 2 selfref_type ■ bit[11:10] - rank 3 selfref_type **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static **Range Variable[x]:** "(DDRCTL_DDR_EN==1) ? (MEMC_NUM_RANKS*2) : 2" + 3
3			**Reserved Field:** Yes

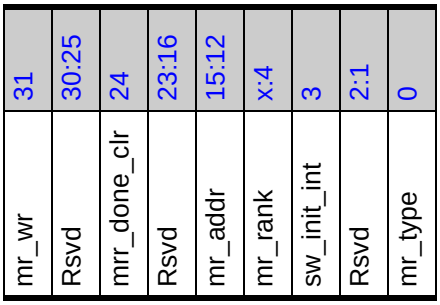
Bits	Name	Memory Access	Description
2:0	operating_mode	R	Operating mode. **DDR4/DDR5 designs:** ■ 000 - Init ■ 001 - Normal ■ 010 - Power-down (For DDR4, this means all ranks are in power-down state. For DDR5, this means at least one rank is in power-down state, check powerdown_state for details) ■ 011 - Self refresh (For DDR4/DDR5, this means all ranks are in self refresh state, check selfref_type for details) ■ 1XX - Maximum Power Saving Mode (For DDR4 only) **LPDDR4/LPDDR5designs:** ■ 000 - Init ■ 001 - Normal ■ 010 - Power-down ■ 011 - Self refresh / Self refresh power-down **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.3.3 MRCTRL0

- **Description:** Mode Register Read/Write Control Register 0.
- **Size:** 32 bits
- **Offset:** 0x11080
- **Exists:** Always

This register is in block REGB_DDRC_CH1.
Note: Do not enable more than one of the following fields simultaneously:

- sw_init_int
- pda_en (DDR4 only field)
- mpr_en (DDR4 only field)
- ppr_en (DDR4 only field)
- ppr_pgmpst_en (DDR4 only field)



31	30:25	24	23:16	15:12	x:4	3	2:1	0
mr_wr	Rsvd	mrr_done_clr	Rsvd	mr_addr	mr_rank	sw_init_int	Rsvd	mr_type

Table 1-184 Fields for Register: MRCTRL0

Bits	Name	Memory Access	Description
31	mr_wr	R/W1S	Setting this register bit to 1 triggers a mode register read or write operation. When the MR operation is complete, the DDRCTL automatically clears this bit. The other register fields of this register must be written in a separate APB transaction, before setting this mr_wr bit. It is recommended NOT to set this signal if in Init, Deep power-down or MPSM operating modes. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
30:25			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
24	mrr_done_clr	R/W1C	If this bit is set, mrr_done will be cleared by the controller. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Testable:** readOnly **Programming Mode:** Dynamic
23:16			**Reserved Field:** Yes
15:12	mr_addr	R/W	Address of the mode register that is to be written to. ■ 0000 - MR0 ■ 0001 - MR1 ■ 0010 - MR2 ■ 0011 - MR3 ■ 0100 - MR4 ■ 0101 - MR5 ■ 0110 - MR6 ■ 0111 - MR7 This signal is also used for writing to control words of the register chip on RDIMMs/LRDIMMs. In that case, it corresponds to the bank address bits sent to the RDIMM/LRDIMM. In case of DDR4, the bit[3:2] corresponds to the bank group bits. Therefore, the bit[3] as well as the bit[2:0] must be set to an appropriate value which is considered both the Address Mirroring of UDIMMs/RDIMMs/LRDIMMs and the Output Inversion of RDIMMs/LRDIMMs. Don't Care for LPDDR4/5 (see MRCTRL1.mr_data for mode register addressing in LPDDR4/5). Don't Care for DDR5 (see CMDCTL.cmd_ctrl for MRW/MRR access in DDR5). **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
x:4	mr_rank	R/W	Controls which rank is accessed by MRCTRL0.mr_wr. Normally, it is desired to access all ranks, so all bits must be set to 1. However, for multi-rank UDIMMs/RDIMMs/LRDIMMs which implement address mirroring, it may be necessary to access ranks individually. Examples (assume DDRCTL is configured for 4 ranks): ■ 0x1 - select rank 0 only ■ 0x2 - select rank 1 only ■ 0x5 - select ranks 0 and 2 ■ 0xA - select ranks 1 and 3 ■ 0xF - select ranks 0, 1, 2 and 3 Don't Care for DDR5. **Value After Reset:** "(MEMC_NUM_RANKS==4) ? 0xF :((MEMC_NUM_RANKS==2) ? 0x3 : 0x1)" **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_NUM_RANKS" + 3
3	sw_init_int	R/W	Indicates whether Software intervention is allowed via MRCTRL0/MRCTRL1 before automatic SDRAM initialization routine or not. For DDR4, this bit can be used to initialize the DDR4 RCD (MR7) before automatic SDRAM initialization. For LPDDR4/5, this bit can be used to program additional mode registers before automatic SDRAM initialization if necessary. In LPDDR4 dual channel mode, note that this must be programmed to both channels beforehand. Note that this must be cleared to 0 after completing Software operation. Otherwise, SDRAM initialization routine will not re-start. ■ 0 - Software intervention is not allowed ■ 1 - Software intervention is allowed Don't Care for DDR5. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
2:1			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
0	mr_type	R/W	Indicates whether the mode register operation is read or write. ■ 0 - Write ■ 1 - Read Only used for LPDDR4/LPDDR5/DDR4. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.3.4 MRCTRL1

- **Description:** Mode Register Read/Write Control Register 1
- **Size:** 32 bits
- **Offset:** 0x11084
- **Exists:** Always

This register is in block REGB_DDRC_CH1.

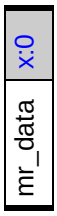


Table 1-185 Fields for Register: MRCTRL1

Bits	Name	Memory Access	Description
x:0	mr_data	R/W	Mode register write data for DDR4 mode. For LPDDR4/5, MRCTRL1[15:0] are interpreted as ■ [15:8] MR Address ■ [7:0] MR data for writes, don't care for read This is 18-bit wide in configurations with DDR4 support and 16-bits for the LPDDR5/4 controller. For DDR4 PPR, this is used for row address field in the ACT command of the PPR sequence. Don't Care for DDR5 (see CMDCTL.cmd_ctrl for MRW access in DDR5). **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_PAGE_BITS" - 1

1.3.5 MRSTAT

- **Description:** Mode Register Read/Write Status Register
- **Size:** 32 bits
- **Offset:** 0x11090
- **Exists:** Always

This register is in block REGB_DDRC_CH1.

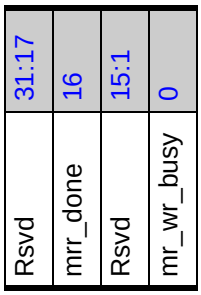


Table 1-186 Fields for Register: MRSTAT

Bits	Name	Memory Access	Description
31:17			**Reserved Field:** Yes
16	mrr_done	R	This signal goes high when the controller received MRR data which is triggered by MRCTRL0.mr_wr. This signal is cleared by mrr_done_clr **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic
15:1			**Reserved Field:** Yes
0	mr_wr_busy	R	The SoC core may initiate a MR write operation only if this signal is low. This signal goes high in the clock after the DDRCTL accepts the MRW/MRR request. It goes low when the MRW/MRR command is issued to the SDRAM. It is recommended not to perform MRW/MRR commands when 'MRSTAT.mr_wr_busy' is high. ■ 0 - Indicates that the SoC core can initiate a mode register write operation ■ 1 - Indicates that mode register write operation is in progress **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.3.6 MRRDATA0

- **Description:** Mode Register Read Data 0
- **Size:** 32 bits
- **Offset:** 0x11094
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_DDRC_CH1.

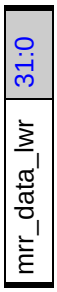


Table 1-187 Fields for Register: MRRDATA0

Bits	Name	Memory Access	Description
31:0	mrr_data_lwr	R	MRR data for DQ[31:0] This register is updated when the controller issued MRR command triggered by MRCTRL register. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic

1.3.7 MRRDATA1

- **Description:** Mode Register Read Data 1
- **Size:** 32 bits
- **Offset:** 0x11098
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_DDRC_CH1.

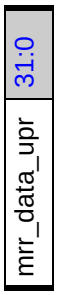


Table 1-188 Fields for Register: MRRDATA1

Bits	Name	Memory Access	Description
31:0	mrr_data_upr	R	MRR data for DQ[63:32] This register is updated when the controller issued MRR command triggered by MRCTRL register. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic

1.3.8 DERATECTL5

- **Description:** Temperature Derate Control Register 5
- **Size:** 32 bits
- **Offset:** 0x11114
- **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1

This register is in block REGB_DDRC_CH1.

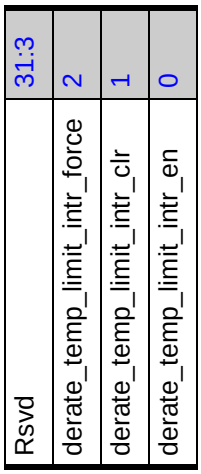


Table 1-189 Fields for Register: DERATECTL5

Bits	Name	Memory Access	Description
31:3			**Reserved Field:** Yes
2	derate_temp_limit_intr_force	R/W1C	Interrupt force bit for derate_temp_limit_intr. Setting this register to 1 will cause the derate_temp_limit_intr output pin to be asserted. At the end of the interrupt force operation, the DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Testable:** readOnly **Programming Mode:** Dynamic
1	derate_temp_limit_intr_clr	R/W1C	Interrupt clear bit for derate_temp_limit_intr. At the end of the interrupt clear operation, the DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
0	derate_temp_limit_intr_en	R/W	Interrupt enable bit for derate_temp_limit_intr output pin. ■ 1 Enabled ■ 0 Disabled **Value After Reset:** 0x1 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Programming Mode:** Dynamic

1.3.9 DERATESTAT0

- **Description:** Temperature Derate Status Register 0
- **Size:** 32 bits
- **Offset:** 0x1111c
- **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1

This register is in block REGB_DDRC_CH1.

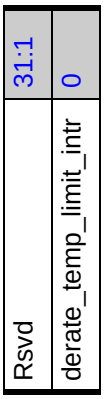


Table 1-190 Fields for Register: DERATESTAT0

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	derate_temp_limit_intr	R	Derate temperature interrupt indicating SDRAM temperature operating limit is exceeded. In LPDDR4, this register field is set to 1 when the value read from MR4[2:0] is 3'b000 or 3'b111. In LPDDR5, this register field is set to 1 when the value read from MR4[4:0] is 5'b00000 or 5'b11111 or invalid value. In DDR5, this register field is set to 1 when the value read from MR4[2:0] is the thresholds programmed by DERATECTL2.derate_low_temp_limit and DERATECTL2.derate_high_temp_limit. Cleared by register DERATECTL1.derate_temp_limit_intr_clr. **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Testable:** readOnly **Programming Mode:** Static

1.3.10 DERATEDBGCTL

- **Description:** Temperature Derate Debug Contrl Register
- **Size:** 32 bits
- **Offset:** 0x11124
- **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1

This register is in block REGB_DDRC_CH1.

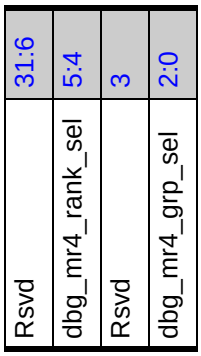


Table 1-191 Fields for Register: DERATEDBGCTL

Bits	Name	Memory Access	Description
31:6			**Reserved Field:** Yes
5:4	dbg_mr4_rank_sel	R/W	MR4 rank select in case of multi ranks **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Programming Mode:** Static
3			**Reserved Field:** Yes
2:0	dbg_mr4_grp_sel	R/W	MR4 data group select based on 4 device MRR read data **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Programming Mode:** Static

1.3.11 DERATEDBGSTAT

- **Description:** Temperature Derate Debug Status Register
- **Size:** 32 bits
- **Offset:** 0x11128
- **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1

This register is in block REGB_DDRC_CH1.

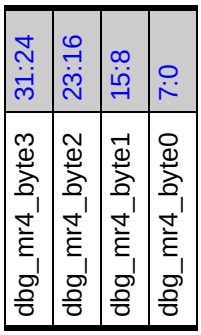


Table 1-192 Fields for Register: DERATEDBGSTAT

Bits	Name	Memory Access	Description
31:24	dbg_mr4_byte3	R	Byte 3 of selected 32 bits MR4 data. LPDDR4/LPDDR5 use 5 bits and DDR5 uses 8 bits. This register is updated when the controller issues MRR for MR4 automatically. For LPDDR4 and LPDDR5 : It is occurred only if derating is enabled. (DERATECTL0.derate_enable=1) **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Programming Mode:** Dynamic
23:16	dbg_mr4_byte2	R	Byte 2 of selected 32 bits MR4 data. LPDDR4/LPDDR5 use 5 bits and DDR5 uses 8 bits. This register is updated when the controller issues MRR for MR4 automatically. For LPDDR4 and LPDDR5 : It is occurred only if derating is enabled. (DERATECTL0.derate_enable=1) **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Programming Mode:** Dynamic
15:8	dbg_mr4_byte1	R	Byte 1 of selected 32 bits MR4 data. LPDDR4/LPDDR5 use 5 bits and DDR5 uses 8 bits. This register is updated when the controller issues MRR for MR4 automatically. For LPDDR4 and LPDDR5 : It is occurred only if derating is enabled. (DERATECTL0.derate_enable=1) **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
7:0	dbg_mr4_byte0	R	Byte 0 of selected 32 bits MR4 data. LPDDR4/LPDDR5 use 5 bits and DDR5 uses 8 bits. This register is updated when the controller issues MRR for MR4 automatically. For LPDDR4 and LPDDR5 : It is occurred only if derating is enabled. (DERATECTL0.derate_enable=1) **Value After Reset:** 0x0 **Exists:** DDRCTL_DDR_OR_MEMC_LPDDR4==1 **Programming Mode:** Dynamic

1.3.12 PWRCTL

- **Description:** Low Power Control Register
- **Size:** 32 bits
- **Offset:** 0x11180
- **Exists:** Always

This register is in block REGB_DDRC_CH1.

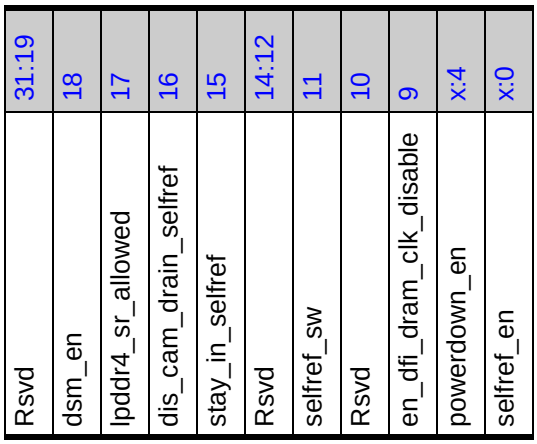


Table 1-193 Fields for Register: PWRCTL

Bits	Name	Memory Access	Description
31:19			**Reserved Field:** Yes
18	dsm_en	R/W	A value of 1 to this register causes system to move to Deep Sleep Mode state immediately. ■ 1 - Entry to Deep Sleep Mode ■ 0 - Exit from Deep Sleep Mode **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic
17	lpddr4_sr_allowed	R/W	Indicates whether transition from SR-PD to SR and back to SR-PD is allowed. This register should be set to '1' if any of PHYMSTR or PPT features is enabled. This register field cannot be modified while PWRCTL.selfref_sw==1. ■ 0 - SR-PD -> SR -> SR-PD not allowed ■ 1 - SR-PD -> SR -> SR-PD allowed **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
16	dis_cam_drain_selfref	R/W	Indicates whether skipping CAM draining is allowed when entering Self-Refresh. This register field cannot be modified while PWRCTL.selfref_sw==1. ■ 0 - CAMs must be empty before entering SR ■ 1 - CAMs are not emptied before entering SR (unsupported) Note, PWRCTL.dis_cam_drain_selfref=1 is unsupported in this release. PWRCTL.dis_cam_drain_selfref=0 is required. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
15	stay_in_selfref	R/W	Self refresh state is an intermediate state to enter to Self refresh power down state or exit Self refresh power down state for LPDDR4/5. This register controls transition from the Self refresh state. ■ 1 - Prohibit transition from Self refresh state ■ 0 - Allow transition from Self refresh state **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Dynamic
14:12			**Reserved Field:** Yes
11	selfref_sw	R/W	A value of 1 to this register causes system to move to Self Refresh state immediately, as long as it is not in INIT or DPD/MPSM operating mode. This is referred to as Software Entry/Exit to Self Refresh. ■ 1 - Software Entry to Self Refresh ■ 0 - Software Exit from Self Refresh **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
10			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
9	en_dfi_dram_clk_disable	R/W	Enable the assertion of dfi_dram_clk_disable whenever a clock is not required by the SDRAM. If set to 0, dfi_dram_clk_disable is never asserted. Assertion of dfi_dram_clk_disable is as follows: In DDR4, can be asserted in following: ■ in Self Refresh ■ in Maximum Power Saving Mode In LPDDR4/LPDDR5, can be asserted in following: ■ in Self Refresh Power Down ■ in Power Down ■ during Normal operation (Clock Stop) In DDR5, can be asserted in following: ■ in Self Refresh In DDR5 (L)RDIMM, the value of this field need to be same as DIMMCTL.dimm_selfref_clock_stop_mode. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
x:4	powerdown_en	R/W	If true then the DDRCTL goes into power-down after a programmable number of cycles "maximum idle clocks before power down" (PWRTMG.powerdown_to_x32). This register bit may be re-programmed during the course of normal operation. For LPDDR4/5 and DDR4, only bit[4] is used. For DDR5, powerdown per rank enable is supported. ■ bit[4] - rank 0 powerdown_en ■ bit[5] - rank 1 powerdown_en ■ bit[6] - rank 2 powerdown_en ■ bit[7] - rank 3 powerdown_en **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "(DDRCTL_DDR_EN==1) ? MEMC_NUM_RANKS : 1" + 3

Bits	Name	Memory Access	Description
x:0	selfref_en	R/W	If true then the DDRCTL puts the SDRAM per rank into Self Refresh after a programmable number of cycles "maximum idle clocks before Self Refresh (PWRTMG.selfref_to_x32)". This register bit may be re-programmed during the course of normal operation. For LPDDR4/5 and DDR4, only bit[0] is used. For DDR5, self-refresh per rank enable is provided. Current self-refresh need to be enabled for all ranks. For DDR5 (L)RDIMM, self-refresh need to be enabled for all ranks of both channels. ■ bit[0] - rank 0 selfref_en ■ bit[1] - rank 1 selfref_en ■ bit[2] - rank 2 selfref_en ■ bit[3] - rank 3 selfref_en **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "(DDRCTL_DDR_EN==1) ? MEMC_NUM_RANKS : 1" - 1

1.3.13 HWLPCTL

- **Description:** Hardware Low Power Control Register
- **Size:** 32 bits
- **Offset:** 0x11184
- **Exists:** Always

This register is in block REGB_DDRC_CH1.

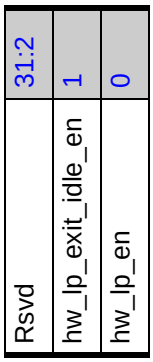


Table 1-194 Fields for Register: HWLPCTL

Bits	Name	Memory Access	Description
31:2			**Reserved Field:** Yes
1	hw_lp_exit_idle_en	R/W	When this bit is programmed to 1 the cactive_in_ddrc pin of the DDRC can be used to exit from the automatic clock stop, automatic power down or automatic self-refresh modes. Note, it will not cause exit of Self-Refresh that was caused by Hardware Low Power Interface and/or Software (PWRCTL.selfref_sw). **Value After Reset:** 0x1 **Exists:** Always **Testable:** readOnly **Programming Mode:** Static
0	hw_lp_en	R/W	Enable for Hardware Low Power Interface. **Value After Reset:** 0x1 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2

1.3.14 ZQCTL1

- **Description:** ZQ Control Register 1
- **Size:** 32 bits
- **Offset:** 0x11284
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_DDRC_CH1.

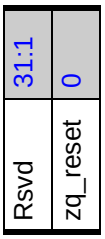


Table 1-195 Fields for Register: ZQCTL1

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	zq_reset	R/W1S	Setting this register bit to 1 triggers a ZQ Reset operation. When the ZQ Reset operation is complete, the DDRCTL automatically clears this bit. It is recommended NOT to set this register bit if in Init, in SR-Powerdown or Deep Sleep Modes. For SR-Powerdown it will be scheduled after SRPD has been exited. For Deep Sleep Mode, it will be scheduled after DSM and/or SRPD has been exited. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Testable:** readOnly **Programming Mode:** Dynamic

1.3.15 ZQSTAT

- **Description:** ZQ Status Register
- **Size:** 32 bits
- **Offset:** 0x1128c
- **Exists:** DDRCTL_LPDDR==1

This register is in block REGB_DDRC_CH1.

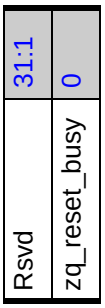


Table 1-196 Fields for Register: ZQSTAT

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	zq_reset_busy	R	SoC core may initiate a ZQ Reset operation only if this signal is low. This signal goes high in the clock after the DDRCTL accepts the ZQ Reset request. It goes low when the ZQ Reset command is issued to the SDRAM and the associated NOP period is over. It is recommended not to perform ZQ Reset commands when this signal is high. ■ 0 - Indicates that the SoC core can initiate a ZQ Reset operation ■ 1 - Indicates that ZQ Reset operation is in progress **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.3.16 DQSOSCSTAT0

- **Description:** DQS/WCK Oscillator Status Register 0
- **Size:** 32 bits
- **Offset:** 0x11304
- **Exists:** LPDDR45_DQSOSC_EN==1

This register is in block REGB_DDRC_CH1.

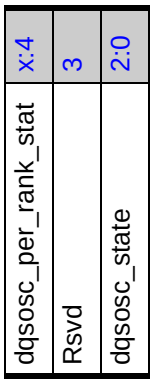


Table 1-197 Fields for Register: DQSOSCSTAT0

Bits	Name	Memory Access	Description
x:4	dqsosc_per_rank_stat	R	DQS/WCK Oscillator per rank status. This bit is set to 0 when DQSOSCCTL0.dqsosc_enable is set to 1, and set to 1 when the DQS Oscillator command sequence is started for the corresponding active rank. **Value After Reset:** 0x0 **Exists:** LPDDR45_DQSOSC_EN==1 **Programming Mode:** Static **Range Variable[x]:** MEMC_NUM_RANKS + 3
3			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
2:0	dqsosc_state	R	DQS/WCK Oscillator Control State Status. ■ 000 - DQSOSC_IDLE ■ 001 - DQSOSC_START: Sending MPC ■ 010 - DQSOSC_RUNTIME: Waiting for runtime passed ■ 011 - DQSOSC_GET_RESULT1: Sending first MRR ■ 100 - DQSOSC_WAIT1: Waiting for tMRR for sending next MRR ■ 101 - DQSOSC_GET_RESULT2: Sending second MRR ■ 110 - DQSOSC_WAIT2: Waiting for tMRR or rank gap The value 0 indicates nothing is being done for DQS Oscillator. Otherwise, DQS Oscillator is running and reflects the current state of DQSOSC. It can be used for debug only to ascertain the current state of DQSOSC controller if it is stuck to one particular state. **Value After Reset:** 0x0 **Exists:** LPDDR45_DQSOSC_EN==1 **Programming Mode:** Static

1.3.17 HWFFCSTAT

- **Description:** Hardware Fast Frequency Change (HWFFC) Status Register
- **Size:** 32 bits
- **Offset:** 0x11404
- **Exists:** UMCTL2_HWFFC_EN==1

This register is in block REGB_DDRC_CH1.

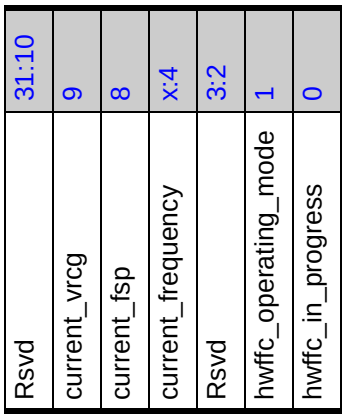


Table 1-198 Fields for Register: HWFFCSTAT

Bits	Name	Memory Access	Description
31:10			**Reserved Field:** Yes
9	current_vrcg	R	Indicates current value of VRCG (MR13 OP[3]). **Value After Reset:** 0x1 **Exists:** UMCTL2_HWFFC_EN==1 **Programming Mode:** Dynamic
8	current_fsp	R	Indicates current value of FSP-OP (MR13 OP[7]). **Value After Reset:** 0x1 **Exists:** UMCTL2_HWFFC_EN==1 && DDRCTL_LPDDR==1 **Programming Mode:** Dynamic
x:4	current_frequency	R	Indicates the current frequency. ■ 0 - Frequency 0/Normal ■ 1 - Frequency 1/FREQ1 ■ 2 - Frequency 2/FREQ2 ■ 3 - Frequency 3/FREQ3 **Value After Reset:** 0x0 **Exists:** UMCTL2_HWFFC_EN==1 **Programming Mode:** Dynamic **Range Variable[x]:** "DDRCTL_FREQUENCY_BITS" + 3

Bits	Name	Memory Access	Description
3:2			**Reserved Field:** Yes
1	hwffc_operating_mode	R	Operating mode of HWFFC. ■ 0 - Normal ■ 1 - Self Refresh or SR-Powerdown **Value After Reset:** 0x0 **Exists:** UMCTL2_HWFFC_EN==1 **Programming Mode:** Dynamic
0	hwffc_in_progress	R	Indicates HWFFC is in progress. **Value After Reset:** 0x0 **Exists:** UMCTL2_HWFFC_EN==1 **Programming Mode:** Dynamic

1.3.18 DFISTAT

- **Description:** DFI Status Register
- **Size:** 32 bits
- **Offset:** 0x11514
- **Exists:** Always

This register is in block REGB_DDRC_CH1.

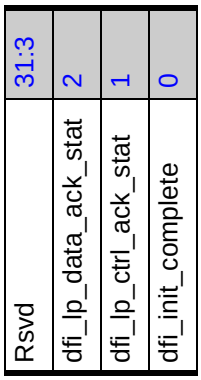


Table 1-199 Fields for Register: DFISTAT

Bits	Name	Memory Access	Description
31:3			**Reserved Field:** Yes
2	dfi_lp_data_ack_stat	R	Stores the value of the dfi_lp_data_ack input to the controller. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
1	dfi_lp_ctrl_ack_stat	R	Stores the value of the dfi_lp_ctrl_ack input to the controller. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
0	dfi_init_complete	R	The status flag register which announces when the DFI initialization has been completed. The DFI INIT triggered by dfi_init_start signal and then the dfi_init_complete flag is polled to know when the initialization is done. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.3.19 DFI0MSGCTL0

- **Description:** DFI0 Message Control Register 0.
- **Size:** 32 bits
- **Offset:** 0x11520
- **Exists:** DDRCTL_DFI_CTRLMSG==1

This register is in block REGB_DDRC_CH1.

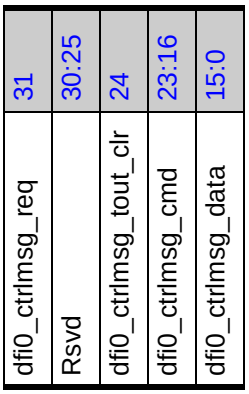


Table 1-200 Fields for Register: DFI0MSGCTL0

Bits	Name	Memory Access	Description
31	dfi0_ctrlmsg_req	R/W1S	Setting this register bit to 1 triggers a DFI controller message transmission operation. DDRCTL automatically clear this bit when the DFI controller message request (dfi0_ctrlmsg_req) is asserted at the DFI MC to PHY Message port interface . This bit must be programmed separately after programming other register fields appropriately of this register. Note: ■ DFI controller message request can be issued only if DFIPHYMSTR.dfi_phymstr_en = 1 ■ DFI controller message request must not be set during DFI LP mode due to software controlled low power entry. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
30:25			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
24	dfi0_ctrlmsg_tout_clr	R/W1C	If this bit is set, DFI0MSGSTAT0.dfi0_ctrlmsg_resp_tout is cleared by the controller. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
23:16	dfi0_ctrlmsg_cmd	R/W	DFI0 controller message command. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
15:0	dfi0_ctrlmsg_data	R/W	DFI0 controller message data. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.3.20 DFI0MSGSTAT0

- **Description:** DFI0 Message Status Register 0
- **Size:** 32 bits
- **Offset:** 0x11524
- **Exists:** DDRCTL_DFI_CTRLMSG==1

This register is in block REGB_DDRC_CH1.

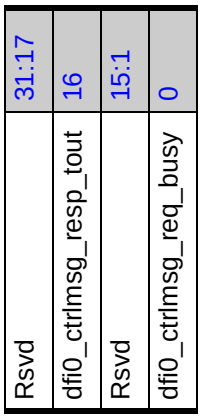


Table 1-201 Fields for Register: DFI0MSGSTAT0

Bits	Name	Memory Access	Description
31:17			**Reserved Field:** Yes
16	dfi0_ctrlmsg_resp_tout	R	This bit is set if dfi0_ctrlmsg_ack is not asserted by PHY within dfi_t_ctrlmsg_resp after asserting dfi0_ctrlmsg_req **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
15:1			**Reserved Field:** Yes
0	dfi0_ctrlmsg_req_busy	R	The SoC must trigger DFI controller message request only if this signal is low. This signal goes high in the clock after the DDRCTL accepts software triggered DFI controller message request by writing into DFI0MSGCTRL0.dfi0_ctrlmsg_req. It goes low when PHY deasserts dfi0_ctrlmsg_ack or dfi0_ctrlmsg_resp_tout event has triggered. ■ 0 - Indicates that the SoC core can initiate a DFI controller message request operation ■ 1 - Indicates that DFI controller message request operation is in progress **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.3.21 ECCSTAT

- **Description:** SECDED ECC Status Register
- **Size:** 32 bits
- **Offset:** 0x11608
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH1.
Valid only in MEMC_ECC_SUPPORT==1 or 3 (SECDED ECC mode)

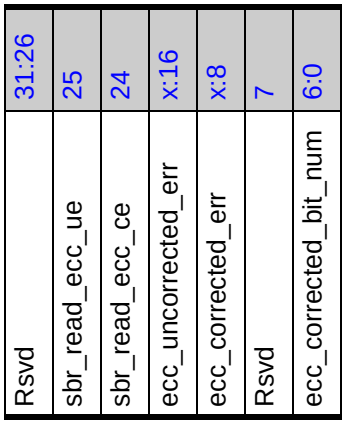


Table 1-202 Fields for Register: ECCSTAT

Bits	Name	Memory Access	Description
31:26			**Reserved Field:** Yes
25	sbr_read_ecc_ue	R	Indicates the uncorrectable error interrupt is due to read operation by scrubber. This bit is cleared on ECCCTL.ecc_uncorrected_err_clr ■ 0 - Mainline/Demand read uncorrectable error interrupt ■ 1 - Scrubber read uncorrectable error interrupt **Value After Reset:** 0x0 **Exists:** UMCTL2_SBR_EN_1==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
24	sbr_read_ecc_ce	R	Indicates the correctable error interrupt is due to read operation by scrubber. This bit is cleared on ECCCTL.ecc_corrected_err_clr ■ 0 - Mainline/Demand read correctable error interrupt ■ 1 - Scrubber read correctable error interrupt **Value After Reset:** 0x0 **Exists:** UMCTL2_SBR_EN_1==1 **Programming Mode:** Static
x:16	ecc_uncorrected_err	R	Double-bit error indicator. In sideband ECC mode, 1 bit per ECC lane. In inline ECC mode, the register always is 1 bit to indicate uncorrectable error on any lane. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Static **Range Variable[x]:** "(MEMC_INLINE_ECC_EN==1 && MEMC_SIDEBAND_ECC_EN==0) ? 1 : (MEMC_FREQ_RATIO==4) ? 8 : 4" + 15
x:8	ecc_corrected_err	R	Single-bit error indicator. In sideband ECC mode, 1 bit per ECC lane. In inline ECC mode, the register always is 1 bit to indicate correctable error on any lane. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Static **Range Variable[x]:** "(MEMC_INLINE_ECC_EN==1 && MEMC_SIDEBAND_ECC_EN==0) ? 1 : (MEMC_FREQ_RATIO==4) ? 8 : 4" + 7
7			**Reserved Field:** Yes
6:0	ecc_corrected_bit_num	R	Bit number corrected by single-bit ECC error. See ECC section of architecture chapter for encoding of this field. If more than one data lane has an error, the lower data lane is selected. This register is 7 bits wide in order to handle 72 bits of the data present in a single lane. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Static

1.3.22 ECCCTL

- **Description:** ECC Clear Register
- **Size:** 32 bits
- **Offset:** 0x1160c
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH1.

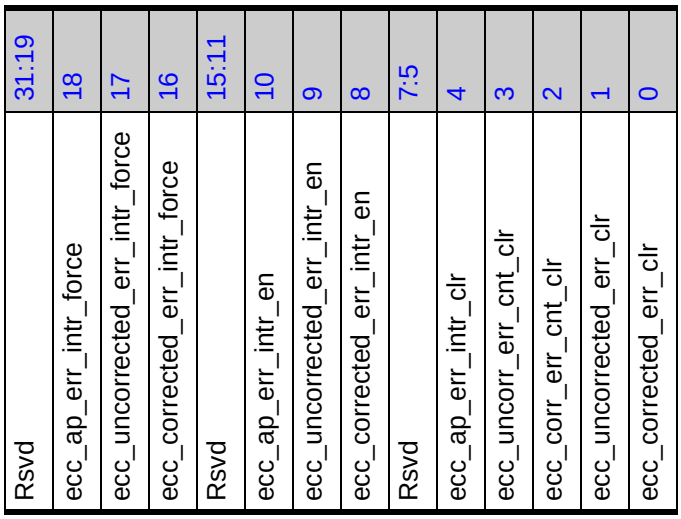


Table 1-203 Fields for Register: ECCCTL

Bits	Name	Memory Access	Description
31:19			**Reserved Field:** Yes
18	ecc_ap_err_intr_force	R/W1C	Interrupt force bit for ecc_ap_err_intr. Setting this register will cause the output interrupt to be asserted. DDRCTL automatically clears this bit. There is no interaction between functionally triggering an interrupt and forcing an interrupt (they are mutually exclusive). **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 && MEMC_ECCAP==1 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
17	ecc_uncorrected_err_intr_force	R/W1C	Interrupt force bit for ecc_uncorrected_err_intr. Setting this register will cause the output interrupt to be asserted. DDRCTL automatically clears this bit. There is no interaction between functionally triggering an interrupt and forcing an interrupt (they are mutually exclusive). **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Testable:** readOnly **Programming Mode:** Dynamic
16	ecc_corrected_err_intr_force	R/W1C	Interrupt force bit for ecc_corrected_err_intr. Setting this register will cause the output interrupt to be asserted. DDRCTL automatically clears this bit. There is no interaction between functionally triggering an interrupt and forcing an interrupt (they are mutually exclusive). **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Testable:** readOnly **Programming Mode:** Dynamic
15:11			**Reserved Field:** Yes
10	ecc_ap_err_intr_en	R/W	Interrupt enable bit for ecc_ap_err_intr. ■ 1: Enabled ■ 0: Disabled **Value After Reset:** 0x1 **Exists:** MEMC_ECC_SUPPORT>0 && MEMC_ECCAP==1 **Programming Mode:** Dynamic
9	ecc_uncorrected_err_intr_en	R/W	Interrupt enable bit for ecc_uncorrected_err_intr. ■ 1: Enabled ■ 0: Disabled **Value After Reset:** 0x1 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic
8	ecc_corrected_err_intr_en	R/W	Interrupt enable bit for ecc_corrected_err_intr. ■ 1 Enabled ■ 0 Disabled **Value After Reset:** 0x1 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic
7:5			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
4	ecc_ap_err_intr_clr	R/W1C	Interrupt clear bit for ecc_ap_err. If this bit is set, the ECCAPSTAT.ecc_ap_err/ecc_ap_err_intr will be cleared. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 && MEMC_ECCAP==1 **Testable:** readOnly **Programming Mode:** Dynamic
3	ecc_uncorr_err_cnt_clr	R/W1C	Setting this register bit to 1 clears the currently stored uncorrected ECC error count. The ECCERRCNT.ecc_uncorr_err_cnt register is cleared by this operation. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Testable:** readOnly **Programming Mode:** Dynamic
2	ecc_corr_err_cnt_clr	R/W1C	Setting this register bit to 1 clears the currently stored corrected ECC error count. The ECCERRCNT.ecc_corr_err_cnt register is cleared by this operation. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Testable:** readOnly **Programming Mode:** Dynamic
1	ecc_uncorrected_err_clr	R/W1C	Setting this register bit to 1 clears the currently stored uncorrected ECC error. The following registers are cleared: ■ ECCSTAT.ecc_uncorrected_err ■ ECCSTAT.sbr_read_ecc_ue ■ ADVECCSTAT.sbr_read_advecc_ue ■ ADVECCSTAT.advecc_uncorr_err_kbd_stat ■ ADVECCSTAT.advecc_uncorrected_err ■ ECCUSYN0 ■ ECCUSYN1 ■ ECCUSYN2 ■ ECCUDATA0 ■ ECCUDATA1 ■ ECCSYMBOL.ecc_uncorr_sym_71_64 DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
0	ecc_corrected_err_clr	R/W1C	Setting this register bit to 1 clears the currently stored corrected ECC error. The following registers are cleared: ■ ECCSTAT.ecc_corrected_err ■ ECCSTAT.sbr_read_ecc_ce ■ ADVECCSTAT.sbr_read_advecc_ce ■ ADVECCSTAT.advecc_corr_err_kbd_stat ■ ADVECCSTAT.advecc_corrected_err ■ ADVECCSTAT.advecc_num_err_symbol ■ ADVECCSTAT.advecc_err_symbol_pos ■ ADVECCSTAT.advecc_err_symbol_bits ■ ECCCSYN0 ■ ECCCSYN1 ■ ECCCSYN2 ■ ECCBITMASK0 ■ ECCBITMASK1 ■ ECCBITMASK2 ■ ECCCDATA0 ■ ECCCDATA1 ■ ECCSYMBOL.ecc_corr_sym_71_64 DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Testable:** readOnly **Programming Mode:** Dynamic

1.3.23 ECCERRCNT

- **Description:** ECC Error Counter Register
- **Size:** 32 bits
- **Offset:** 0x11610
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH1.

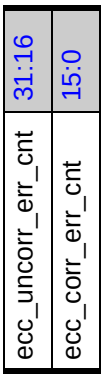


Table 1-204 Fields for Register: ECCERRCNT

Bits	Name	Memory Access	Description
31:16	ecc_uncorr_err_cnt	R	Number of uncorrectable ECC errors detected. It will saturates at 0xFFFF **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic
15:0	ecc_corr_err_cnt	R	Number of correctable ECC errors detected. It will saturates at 0xFFFF **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic

1.3.24 ECCCADDR0

- **Description:** ECC Corrected Error Address Register 0
- **Size:** 32 bits
- **Offset:** 0x11614
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH1.

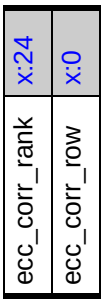


Table 1-205 Fields for Register: ECCCADDR0

Bits	Name	Memory Access	Description
x:24	ecc_corr_rank	R	Indicates the rank number of a read resulting in a corrected ECC error **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 && MEMC_NUM_RANKS>1 **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_RANK_BITS" + 23
x:0	ecc_corr_row	R	Indicates the page/row number of a read resulting in a corrected ECC error. This is 18-bits wide in configurations with LPDDR5 or DDR4 support and 16-bits in all other configurations. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_PAGE_BITS" - 1

1.3.25 ECCCADDR1

- **Description:** ECC Corrected Error Address Register 1
- **Size:** 32 bits
- **Offset:** 0x11618
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH1.

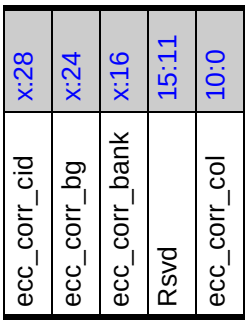


Table 1-206 Fields for Register: ECCCADDR1

Bits	Name	Memory Access	Description
x:28	ecc_corr_cid	R	CID number of a read resulting in a corrected ECC error. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 && UMCTL2_CID_EN==1 **Programming Mode:** Dynamic **Range Variable[x]:** "UMCTL2_CID_WIDTH" + 27
x:24	ecc_corr_bg	R	Bank Group number of a read resulting in a corrected ECC error. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_BG_BITS" + 23
x:16	ecc_corr_bank	R	Bank number of a read resulting in a corrected ECC error. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_BANK_BITS" + 15
15:11			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
10:0	ecc_corr_col	R	Block number of a read resulting in a corrected ECC error (lowest bit not assigned here). **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic

1.3.26 ECCCSYN0

- **Description:** ECC Corrected Syndrome Register 0
- **Size:** 32 bits
- **Offset:** 0x1161c
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH1.

Table 1-207 Fields for Register: ECCCSYN0

Bits	Name	Memory Access	Description
31:0	ecc_corr_syndromes_31_0	R	Data pattern that resulted in a corrected error. For 16-bit ECC, only bits [15:0] are used. This field can be masked by setting the dis_regs_ecc_syndrome input to value 1. This field will be cleared when ecc_corrected_err_clr is set. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic

1.3.27 ECCCSYN1

- **Description:** ECC Corrected Syndrome Register 1
- **Size:** 32 bits
- **Offset:** 0x11620
- **Exists:** MEMC_ECC_SUPPORT>0 && (MEMC_DRAM_DATA_WIDTH_64_OR_MEMC_IN-LINE_ECC==1)

This register is in block REGB_DDRC_CH1.

Table 1-208 Fields for Register: ECCCSYN1

Bits	Name	Memory Access	Description
31:0	ecc_corr_syndromes_63_32	R	Data pattern that resulted in a corrected error. For 32-bit ECC and 16-bit ECC, this register is not used. However, for multi-beat ECC, it represents the data pattern of odd SDRAM data beat (ECC lane). This field can be masked by setting the dis_regs_ecc_syndrome input to value 1.. This field will be cleared when ecc_corrected_err_clr is set. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 && (MEMC_DRAM_DATA_WIDTH_64_OR_MEMC_INLINE_ECC ==1) **Programming Mode:** Dynamic

1.3.28 ECCCSYN2

- **Description:** ECC Corrected Syndrome Register 2
- **Size:** 32 bits
- **Offset:** 0x11624
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH1.

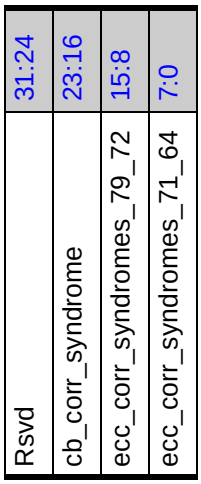


Table 1-209 Fields for Register: ECCCSYN2

Bits	Name	Memory Access	Description
31:24			**Reserved Field:** Yes
23:16	cb_corr_syndrome	R	Indicates the Checkbit corrected error syndrome that resulted in ECC error. It is computed by XOR operation of incoming checkbits and computed checkbits of the incoming data bits. It indicates which bit is in error, or whether multiple bits are in Error. This field will be cleared when ecc_corrected_err_clr is set. **Value After Reset:** 0x0 **Exists:** MEMC_SECDED_SIDEBAND_ECC==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
15:8	ecc_corr_syndromes_79_72	R	Indicates the data pattern that resulted in a corrected error. This register refers to the higher ECC byte, which is bits [79:72] for 64-bit ECC, [47:40] for 32-bit ECC, or [31:24] for 16-bit ECC. This field can be masked by setting the dis_regs_ecc_syndrome input to value 1. This field will be cleared when ecc_corrected_err_clr is set. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT==4 **Programming Mode:** Dynamic
7:0	ecc_corr_syndromes_71_64	R	Indicates the data pattern that resulted in a corrected error. This register refers to the ECC byte, which is bits [71:64] for 64-bit ECC, [39:32] for 32-bit ECC, or [23:16] for 16-bit ECC. This field can be masked by setting the dis_regs_ecc_syndrome input to value 1. This field will be cleared when ecc_corrected_err_clr is set. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic

1.3.29 ECCBITMASK0

- **Description:** ECC Corrected Data Bit Mask Register 0
- **Size:** 32 bits
- **Offset:** 0x11628
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH1.

Table 1-210 Fields for Register: ECCBITMASK0

Bits	Name	Memory Access	Description
31:0	ecc_corr_bit_mask_31_0	R	Mask for the corrected data portion ■ 1 on any bit indicates that the bit has been corrected by the ECC logic ■ 0 on any bit indicates that the bit has not been corrected by the ECC logic This register accumulates data over multiple ECC errors, to give an overall indication of which bits are being fixed. It is cleared by writing a 1 to ECCCTL.ecc_corrected_err_clr. For 16-bit ECC, only bits [15:0] are used **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic

1.3.30 ECCBITMASK1

- **Description:** ECC Corrected Data Bit Mask Register 1
- **Size:** 32 bits
- **Offset:** 0x1162c
- **Exists:** MEMC_ECC_SUPPORT>0 && (MEMC_DRAM_DATA_WIDTH_64_OR_MEMC_INLINE_ECC==1)

This register is in block REGB_DDRC_CH1.

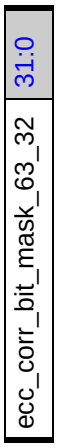


Table 1-211 Fields for Register: ECCBITMASK1

Bits	Name	Memory Access	Description
31:0	ecc_corr_bit_mask_63_32	R	Mask for the corrected data portion ■ 1 on any bit indicates that the bit has been corrected by the ECC logic ■ 0 on any bit indicates that the bit has not been corrected by the ECC logic This register accumulates data over multiple ECC errors, to give an overall indication of which bits are being fixed. It is cleared by writing a 1 to ECCCTL.ecc_corrected_err_clr. For 32-bit ECC and 16-bit ECC, this register is not used. However, for multi-beat ECC, it represents the ECC errors of odd SDRAM data beat (ECC lane). **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 && (MEMC_DRAM_DATA_WIDTH_64_OR_MEMC_INLINE_ECC==1) **Programming Mode:** Dynamic

1.3.31 ECCBITMASK2

- **Description:** ECC Corrected Data Bit Mask Register 2
- **Size:** 32 bits
- **Offset:** 0x11630
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH1.

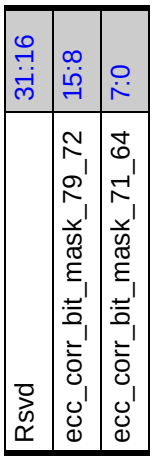


Table 1-212 Fields for Register: ECCBITMASK2

Bits	Name	Memory Access	Description
31:16			**Reserved Field:** Yes
15:8	ecc_corr_bit_mask_79_72	R	Indicates the mask for the corrected data portion. ■ 1 on any bit indicates that the bit has been corrected by the ECC logic ■ 0 on any bit indicates that the bit has not been corrected by the ECC logic This register accumulates data over multiple ECC errors, to give an overall indication of which bits are being fixed. It is cleared by writing a 1 to ECCCTL.ecc_corrected_err_clr. This register refers to the higher ECC byte, which is bits [79:72] for 64-bit ECC, [47:40] for 32-bit ECC, or [31:24] for 16-bit ECC. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT==4 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
7:0	ecc_corr_bit_mask_71_64	R	Mask for the corrected data portion ■ 1 on any bit indicates that the bit has been corrected by the ECC logic ■ 0 on any bit indicates that the bit has not been corrected by the ECC logic This register accumulates data over multiple ECC errors, to give an overall indication of which bits are being fixed. It is cleared by writing a 1 to ECCCTL.ecc_corrected_err_clr. This register refers to the ECC byte, which is bits [71:64] for 64-bit ECC, [39:32] for 32-bit ECC, or [23:16] for 16-bit ECC **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic

1.3.32 ECCUADDR0

- **Description:** ECC Uncorrected Error Address Register 0
- **Size:** 32 bits
- **Offset:** 0x11634
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH1.

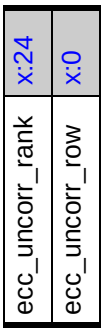


Table 1-213 Fields for Register: ECCUADDR0

Bits	Name	Memory Access	Description
x:24	ecc_uncorr_rank	R	Rank number of a read resulting in an uncorrected ECC error. **Value After Reset:** 0x0 **Exists:** MEMC_NUM_RANKS>1 **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_RANK_BITS" + 23
x:0	ecc_uncorr_row	R	Page/row number of a read resulting in an uncorrected ECC error. This is 18-bits wide in configurations with LPDDR5 or DDR4 support and 16-bits in all other configurations. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_PAGE_BITS" - 1

1.3.33 ECCUADDR1

- **Description:** ECC Uncorrected Error Address Register 1
- **Size:** 32 bits
- **Offset:** 0x11638
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH1.

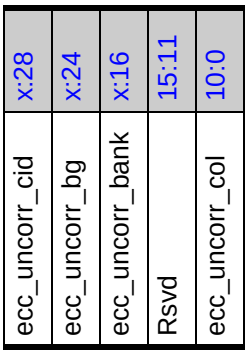


Table 1-214 Fields for Register: ECCUADDR1

Bits	Name	Memory Access	Description
x:28	ecc_uncorr_cid	R	CID number of a read resulting in an uncorrected ECC error. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 && UMCTL2_CID_EN==1 **Programming Mode:** Dynamic **Range Variable[x]:** "UMCTL2_CID_WIDTH" + 27
x:24	ecc_uncorr_bg	R	Bank Group number of a read resulting in an uncorrected ECC error **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_BG_BITS" + 23
x:16	ecc_uncorr_bank	R	Bank number of a read resulting in an uncorrected ECC error **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_BANK_BITS" + 15
15:11			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
10:0	ecc_uncorr_col	R	Block number of a read resulting in an uncorrected ECC error (lowest bit not assigned here) **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic

1.3.34 ECCUSYN0

- **Description:** ECC Uncorrected Syndrome Register 0
- **Size:** 32 bits
- **Offset:** 0x1163c
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH1.

Table 1-215 Fields for Register: ECCUSYN0

Bits	Name	Memory Access	Description
31:0	ecc_uncorr_syndromes_31_0	R	Data pattern that resulted in an uncorrected error. For 16-bit ECC, only bits [15:0] are used. This field can be masked by setting the dis_regs_ecc_syndrome input to value 1. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic

1.3.35 ECCUSYN1

- **Description:** ECC Uncorrected Syndrome Register 1
- **Size:** 32 bits
- **Offset:** 0x11640
- **Exists:** MEMC_ECC_SUPPORT>0 && (MEMC_DRAM_DATA_WIDTH_64_OR_MEMC_INLINE_ECC==1)

This register is in block REGB_DDRC_CH1.

Table 1-216 Fields for Register: ECCUSYN1

Bits	Name	Memory Access	Description
31:0	ecc_uncorr_syndromes_63_32	R	Data pattern that resulted in an uncorrected error. For 32-bit ECC and 16-bit ECC, this register is not used. However, for multi-beat ECC, it represents the data pattern of odd SDRAM data beat (ECC lane). This field can be masked by setting the dis_regs_ecc_syndrome input to value 1. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 && (MEMC_DRAM_DATA_WIDTH_64_OR_MEMC_INLINE_ECC==1) **Programming Mode:** Dynamic

1.3.36 ECCUSYN2

- **Description:** ECC Uncorrected Syndrome Register 2
- **Size:** 32 bits
- **Offset:** 0x11644
- **Exists:** MEMC_ECC_SUPPORT>0

This register is in block REGB_DDRC_CH1.

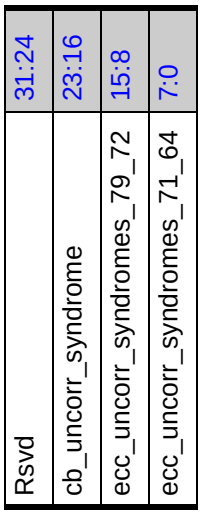


Table 1-217 Fields for Register: ECCUSYN2

Bits	Name	Memory Access	Description
31:24			**Reserved Field:** Yes
23:16	cb_uncorr_syndrome	R	Indicates the Checkbit uncorrected error syndrome that resulted in ECC error. . It is computed by XOR operation of incoming checkbits and computed checkbits of the incoming data bits. It indicates which bit is in error, or whether multiple bits are in Error. **Value After Reset:** 0x0 **Exists:** MEMC_SECDED_SIDEBAND_ECC==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
15:8	ecc_uncorr_syndromes_79_72	R	Data pattern that resulted in an uncorrected error. This register refers to the higher ECC byte, which is bits [79:72] for 64-bit ECC, [47:40] for 32-bit ECC, or [31:24] for 16-bit ECC. This field can be masked by setting the dis_regs_ecc_syndrome input to value 1. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT==4 **Programming Mode:** Dynamic
7:0	ecc_uncorr_syndromes_71_64	R	Data pattern that resulted in an uncorrected error. This register refers to the ECC byte, which is bits [71:64] for 64-bit ECC, [39:32] for 32-bit ECC, or [23:16] for 16-bit ECC. This field can be masked by setting the dis_regs_ecc_syndrome input to value 1. **Value After Reset:** 0x0 **Exists:** MEMC_ECC_SUPPORT>0 **Programming Mode:** Dynamic

1.3.37 ECCAPSTAT

- **Description:** Address protection within ECC Status Register
- **Size:** 32 bits
- **Offset:** 0x11664
- **Exists:** MEMC_ECCAP==1

This register is in block REGB_DDRC_CH1.

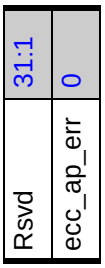


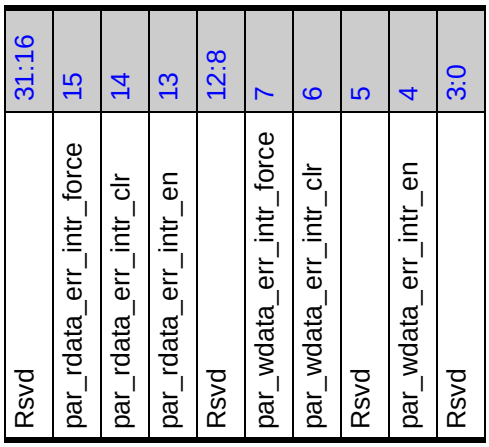
Table 1-218 Fields for Register: ECCAPSTAT

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	ecc_ap_err	R	Indicates the number of ECC errors (correctable/uncorrectable) within one burst exceeded the threshold.(ECCCFG0.ecc_ap_err_threshold) **Value After Reset:** 0x0 **Exists:** MEMC_ECCAP==1 **Programming Mode:** Static

1.3.38 OCPARCFG0

- **Description:** On-Chip Parity Configuration Register 0
- **Size:** 32 bits
- **Offset:** 0x11680
- **Exists:** UMCTL2_OCPAR_OR_OCECC_EN_1==1

This register is in block REGB_DDRC_CH1.



31:16	15	14	13	12:8	7	6	5	4	3:0
Rsvd	par_rdata_err_intr_force	par_rdata_err_intr_clr	par_rdata_err_intr_en	Rsvd	par_wdata_err_intr_force	par_wdata_err_intr_clr	Rsvd	par_wdata_err_intr_en	Rsvd

Table 1-219 Fields for Register: OCPARCFG0

Bits	Name	Memory Access	Description
31:16			**Reserved Field:** Yes
15	par_rdata_err_intr_force	R/W1C	Interrupt force bit for all par_rdata_err_intr_n and par_rdata_in_err_ecc_intr (Inline-ECC only). DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** UMCTL2_OCPAR_EN_1==1 **Testable:** readOnly **Programming Mode:** Dynamic
14	par_rdata_err_intr_clr	R/W1C	Interrupt clear bit for par_rdata_err_intr. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** UMCTL2_OCPAR_EN_1==1 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
13	par_rdata_err_intr_en	R/W	Enables interrupt generation, if set to 1, for all ports, on signal par_rdata_err_intr_n upon detection of parity error at the AXI interface. **Value After Reset:** 0x1 **Exists:** UMCTL2_OCPAR_EN_1==1 **Programming Mode:** Dynamic
12:8			**Reserved Field:** Yes
7	par_wdata_err_intr_force	R/W1C	Interrupt force bit for par_wdata_err_intr. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** UMCTL2_OCPAR_EN_1==1 **Testable:** readOnly **Programming Mode:** Dynamic
6	par_wdata_err_intr_clr	R/W1C	Interrupt clear bit for par_wdata_err_intr. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** UMCTL2_OCPAR_EN_1==1 **Testable:** readOnly **Programming Mode:** Dynamic
5			**Reserved Field:** Yes
4	par_wdata_err_intr_en	R/W	Enables write data interrupt generation (par_wdata_err_intr) upon detection of parity error at the AXI or DFI interface. **Value After Reset:** 0x1 **Exists:** UMCTL2_OCPAR_EN_1==1 **Volatile:** true **Programming Mode:** Dynamic
3:0			**Reserved Field:** Yes

1.3.39 OCPARSTAT2

- **Description:** On-Chip Parity Status Register 2
- **Size:** 32 bits
- **Offset:** 0x11690
- **Exists:** UMCTL2_OCPAR_EN_1==1

This register is in block REGB_DDRC_CH1.

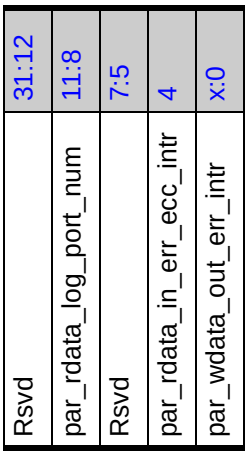


Table 1-220 Fields for Register: OCPARSTAT2

Bits	Name	Memory Access	Description
31:12			**Reserved Field:** Yes
11:8	par_rdata_log_port_num	R	Failing port number (binary encoded) of the last read data beat which resulted in on-chip parity error at the AXI interface. If there are more than one simultaneous port failures, the lower-indexed port is captured. **Value After Reset:** 0x0 **Exists:** MEMC_INLINE_ECC==0 **Programming Mode:** Static
7:5			**Reserved Field:** Yes
4	par_rdata_in_err_ecc_intr	R	Interrupt on ECC data going into inline ECC decoder. Cleared by par_rdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** MEMC_INLINE_ECC==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
x:0	par_wdata_out_err_intr	R	Write data parity error interrupt on output. Cleared by register par_wdata_err_intr_clr. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static **Range Variable[x]:** "UMCTL2_OCPAR_WDATA_OUT_ERR_WIDTH" - 1

1.3.40 OCCAPCFG1

- **Description:** On-Chip command/Address Protection Configuration Register 1
- **Size:** 32 bits
- **Offset:** 0x11788
- **Exists:** UMCTL2_OCCAP_EN_1==1

This register is in block REGB_DDRC_CH1.

31:27	26	25	24	23:19	18	17	16	15:11	10	9	8	7:3	2	1	0
Rsvd	occap_ddrc_ctrl_poison_err_inj	occap_ddrc_ctrl_poison_parallel	occap_ddrc_ctrl_poison_seq	Rsvd	occap_ddrc_ctrl_intr_force	occap_ddrc_ctrl_intr_clr	occap_ddrc_ctrl_intr_en	Rsvd	occap_ddrc_data_poison_err_inj	occap_ddrc_data_poison_parallel	occap_ddrc_data_poison_seq	Rsvd	occap_ddrc_data_intr_force	occap_ddrc_data_intr_clr	occap_ddrc_data_intr_en

Table 1-221 Fields for Register: OCCAPCFG1

Bits	Name	Memory Access	Description
31:27			**Reserved Field:** Yes
26	occap_ddrc_ctrl_poison_err_inj	R/W	Enable error injection in the poisoning of OCCAP DDRC CTRL logic Injects error into poisoning logic (either parallel or seq) such that XOR logic for one signal is not poisoned when expected. If set, it allows ability to corrupt the following register fields. ■ 1'b0: OCCAPSTAT1.occap_ddrc_ctrl_poison_parallel/seq_err=0 ■ 1'b1: OCCAPSTAT1.occap_ddrc_ctrl_poison_parallel/seq_err=1 Do not change value in same APB write as setting of occap_ddrc_ctrl_poison_parallel/_seq **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
25	occap_ddrc_ctrl_poison_parallel	R/W1C	Enables poisoning of OCCAP DDRC CTRL logic for all parts of comparison logic, in parallel. Poisons comparison logic for one core_ddrc_core_clk cycle by inverting a ddrc_ctrl[0]'s signal to XOR logic. ddrc_ctrl[1] related signals are never poisoned. All signals are poisoned in parallel and checks if all signals were poisoned correctly. If this is not the case, this is flagged by OCCAPSTAT1.occap_ddrc_ctrl_poison_parallel_err=1. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
24	occap_ddrc_ctrl_poison_seq	R/W1C	Enables poisoning of OCCAP DDRC CTRL logic for all parts of comparison logic, in sequence. Poisons comparison logic for one core_ddrc_core_clk cycle by inverting a ddrc_ctrl[0]'s signal to XOR logic. ddrc_ctrl[1] related signals are never poisoned. Each signal from ddrc_ctrl[0] is poisoned in series and checks in turn, that each signal was poisoned correctly. If this is not the case, this is flagged by OCCAPSTAT1.occap_ddrc_ctrl_poison_seq_err=1. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
23:19			**Reserved Field:** Yes
18	occap_ddrc_ctrl_intr_force	R/W1C	Interrupt force bit for occap_ddrc_ctrl_err_intr, setting this register will cause the output interrupt to be asserted. DDRCTL automatically clears this bit. There is no interaction between functionally triggering an interrupt and forcing an interrupt (they are mutually exclusive). **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
17	occap_ddrc_ctrl_intr_clr	R/W1C	Interrupt clear bit for occap_ddrc_ctrl_err_intr. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic

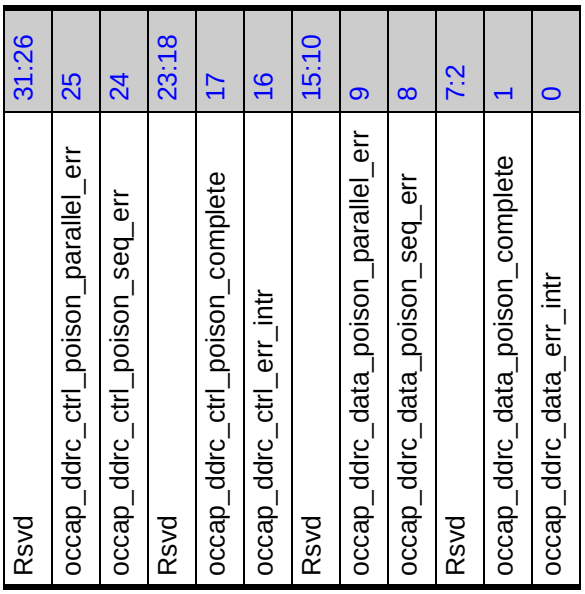
Bits	Name	Memory Access	Description
16	occap_ddrc_ctrl_intr_en	R/W	Enables interrupt generation on signal occap_ddrc_ctrl_err_intr upon detection of OCCAP DDRC CTRL errors. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Dynamic
15:11			**Reserved Field:** Yes
10	occap_ddrc_data_poison_err_inj	R/W	Enable error injection in the poisoning of OCCAP DDRC DATA logic Injects error into poisoning logic (either parallel or seq) such that XOR logic for one signal is not poisoned when expected. If set, it allows ability to corrupt the following register fields. ■ 1'b0: OCCAPSTAT1.occap_ddrc_data_poison_parallel/seq_err=0 ■ 1'b1: OCCAPSTAT1.occap_ddrc_data_poison_parallel/seq_err=1 Do not change value in same APB write as setting of occap_ddrc_data_poison_parallel/_seq **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
9	occap_ddrc_data_poison_parallel	R/W1C	Enables poisoning of OCCAP DDRC DATA logic for all parts of comparison logic, in parallel. Poisons comparison logic for one core_ddrc_core_clk cycle by inverting all bits of a signal to XOR logic. All signals of instance[0] of the duplicated modules are poisoned in parallel and checks if all signals were poisoned correctly. If this is not the case, this is flagged by OCCAPSTAT1.occap_ddrc_data_poison_parallel_err=1. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
8	occap_ddrc_data_poison_seq	R/W1C	Enables poisoning of OCCAP DDRC DATA logic for all parts of comparison logic, in sequence. Poisons comparison logic for one core_ddrc_core_clk cycle by inverting all bits of a signal to XOR logic. All signals of instance[0] of the duplicated modules are poisoned in parallel and checks if all signals were poisoned correctly. If this is not the case, this is flagged by OCCAPSTAT1.occap_ddrc_data_poison_seq_err=1. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
7:3			**Reserved Field:** Yes
2	occap_ddrc_data_intr_force	R/W1C	Interrupt force bit for occap_ddrc_data_err_intr, setting this register will cause the output interrupt to be asserted. DDRCTL automatically clears this bit. There is no interaction between functionally triggering an interrupt and forcing an interrupt (they are mutually exclusive). **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
1	occap_ddrc_data_intr_clr	R/W1C	Interrupt clear bit for occap_ddrc_data_err_intr. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
0	occap_ddrc_data_intr_en	R/W	Enables interrupt generation on signal occap_ddrc_data_err_intr upon detection of OCCAP DDRC DATA errors. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Dynamic

1.3.41 OCCAPSTAT1

- **Description:** On-Chip command/Address Protection Status Register 1
- **Size:** 32 bits
- **Offset:** 0x1178c
- **Exists:** UMCTL2_OCCAP_EN_1==1

This register is in block REGB_DDRC_CH1.



31:26	25	24	23:18	17	16	15:10	9	8	7:2	1	0
Rsvd	occap_ddrc_ctrl_poison_parallel_err	occap_ddrc_ctrl_poison_seq_err	Rsvd	occap_ddrc_ctrl_poison_complete	occap_ddrc_ctrl_err_intr	Rsvd	occap_ddrc_data_poison_parallel_err	occap_ddrc_data_poison_seq_err	Rsvd	occap_ddrc_data_poison_complete	occap_ddrc_data_err_intr

Table 1-222 Fields for Register: OCCAPSTAT1

Bits	Name	Memory Access	Description
31:26			**Reserved Field:** Yes
25	occap_ddrc_ctrl_poison_parallel_err	R	Error when occap_ddrc_ctrl_poison_parallel was active due to incorrect no. of errors being occurring. Internal logic checks the number of errors detected while poisoning occurred for occap_ddrc_ctrl_poison_parallel. It checks for error on all of the the corresponding XOR outputs. If multi-bit, checks also that all XOR bits are set. It checks all XOR in parallel. Register is valid only when occap_ddrc_ctrl_cmp_poison_complete=1. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

Bits	Name	Memory Access	Description
24	occap_ddrc_ctrl_poison_seq_err	R	Error when occap_ddrc_ctrl_poison_seq was active due to incorrect no. of errors being occurring. Internal logic checks the number of errors detected while poisoning occurred for occap_ddrc_ctrl_poison_seq. It checks for error on all of the corresponding XOR outputs. It checks each XOR sequentially. Register is valid only when occap_ddrc_ctrl_cmp_poison_complete=1. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
23:18			**Reserved Field:** Yes
17	occap_ddrc_ctrl_poison_complete	R	OCCAP DDRC CTRL poisoning complete interrupt status. Register cleared by OCCAPCFG1.occap_ddrc_ctrl_err_intr_clr. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
16	occap_ddrc_ctrl_err_intr	R	OCCAP DDRC CTRL error interrupt status. Register cleared by OCCAPCFG1.occap_ddrc_ctrl_err_intr_clr. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
15:10			**Reserved Field:** Yes
9	occap_ddrc_data_poison_parallel_err	R	Error when occap_ddrc_data_poison_parallel was active due to incorrect no. of errors being occurring. Internal logic checks the number of errors detected while poisoning occurred for occap_ddrc_data_poison_parallel. It checks for error on all of the the corresponding XOR outputs. If multi-bit, checks also that all XOR bits are set. It checks all XOR in parallel. This is cleared when OCCAPCFG1.occap_ddrc_data_poison_parallel=0 occurs. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

Bits	Name	Memory Access	Description
8	occap_ddrc_data_poison_seq_err	R	Error when occap_ddrc_data_poison_seq was active due to incorrect no. of errors being occurring. Internal logic checks the number of errors detected while poisoning occurred for occap_ddrc_data_poison_seq. It checks for error on all of the corresponding XOR outputs. It checks each XOR sequentially. This is cleared when OCCAPCFG1.occap_ddrc_data_poison_seq=0 occurs. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:2			**Reserved Field:** Yes
1	occap_ddrc_data_poison_complete	R	OCCAP DDRC DATA poisoning complete interrupt status. Register cleared by OCCAPCFG1.occap_ddrc_data_err_intr_clr. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
0	occap_ddrc_data_err_intr	R	OCCAP DDRC DATA error interrupt status. Register cleared by OCCAPCFG1.occap_ddrc_data_err_intr_clr. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.3.42 OCCAPCFG2

- **Description:** On-Chip command/Address Protection Configuration Register 2
- **Size:** 32 bits
- **Offset:** 0x11790
- **Exists:** UMCTL2_OCCAP_EN_1==1

This register is in block REGB_DDRC_CH1.

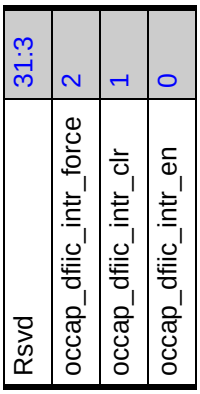


Table 1-223 Fields for Register: OCCAPCFG2

Bits	Name	Memory Access	Description
31:3			**Reserved Field:** Yes
2	occap_dfiic_intr_force	R/W1C	Interrupt force bit for occap_dfiic_err_intr, setting this register will cause the output interrupt to be asserted. DDRCTL automatically clears this bit. There is no interaction between functionally triggering an interrupt and forcing an interrupt (they are mutually exclusive). **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
1	occap_dfiic_intr_clr	R/W1C	Interrupt clear bit for occap_dfiic_err_intr. DDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
0	occap_dfiic_intr_en	R/W	Enables interrupt generation on signal occap_dfiic_err_intr upon detection of OCCAP DFI interconnect errors. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Dynamic

1.3.43 OCCAPSTAT2

- **Description:** On-Chip command/Address Protection Status Register 2
- **Size:** 32 bits
- **Offset:** 0x11794
- **Exists:** UMCTL2_OCCAP_EN_1==1

This register is in block REGB_DDRC_CH1.

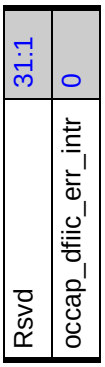


Table 1-224 Fields for Register: OCCAPSTAT2

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	occap_dfiic_err_intr	R	OCCAP DFI interconnect error interrupt status. Register cleared by OCCAPCFG2.occap_dfiic_intr_clr. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.3.44 LNKECCCTL0

- **Description:** Link-ECC Control Register 0
- **Size:** 32 bits
- **Offset:** 0x11980
- **Exists:** MEMC_LINK_ECC==1

This register is in block REGB_DDRC_CH1.

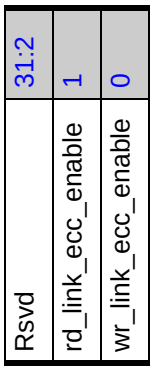


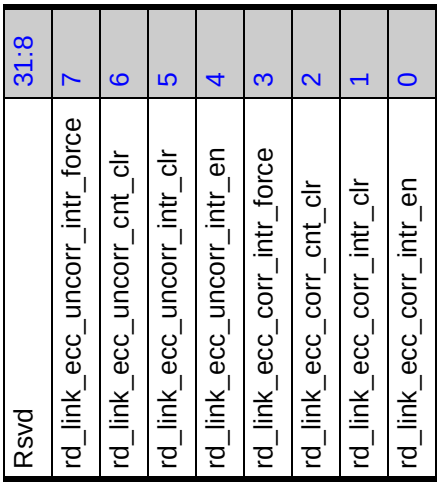
Table 1-225 Fields for Register: LNKECCCTL0

Bits	Name	Memory Access	Description
31:2			**Reserved Field:** Yes
1	rd_link_ecc_enable	R/W	Enable LPDDR5 Read Link ECC feature. ■ 0 - Disabel LPDDR5 Read Link ECC ■ 1 - Enable LPDDR5 Read Link ECC When non-LPDDR5 devices are used, this register must be set to 0. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Volatile:** true **Programming Mode:** Static
0	wr_link_ecc_enable	R/W	Enable LPDDR5 Write Link ECC feature. ■ 0 - Disabel LPDDR5 Write Link ECC ■ 1 - Enable LPDDR5 Write Link ECC When non-LPDDR5 devices are used, this register must be set to 0. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Volatile:** true **Programming Mode:** Static

1.3.45 LNKECCCTL1

- **Description:** Link-ECC Control Register 1
- **Size:** 32 bits
- **Offset:** 0x11984
- **Exists:** MEMC_LINK_ECC==1

This register is in block REGB_DDRC_CH1.
Note: Do not perform any APB access to LNKECCCTL1 within 32 pclk cycles of previous access to LNKECCCTL1, as this might lead to data loss.



31:8	7	6	5	4	3	2	1	0
Rsvd	rd_link_ecc_uncorr_intr_force	rd_link_ecc_uncorr_cnt_clr	rd_link_ecc_uncorr_intr_clr	rd_link_ecc_uncorr_intr_en	rd_link_ecc_corr_intr_force	rd_link_ecc_corr_cnt_clr	rd_link_ecc_corr_intr_clr	rd_link_ecc_corr_intr_en

Table 1-226 Fields for Register: LNKECCCTL1

Bits	Name	Memory Access	Description
31:8			**Reserved Field:** Yes
7	rd_link_ecc_uncorr_intr_force	R/W1C	Interrupt force bit for rd_linkecc_uncorr_err_intr. Setting this register will cause the output interrupt to be asserted. DDRCTL automatically clears this bit. There is no interaction between functionally triggering an interrupt and forcing an interrupt (they are mutually exclusive). **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Testable:** readOnly **Programming Mode:** Dynamic
6	rd_link_ecc_uncorr_cnt_clr	R/W1C	Clear all Read Link-ECC uncorrectable error count. If this bit set,LNKECCERRCNT0.rd_link_ecc_uncorr_cnt will be cleared. LPDDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
5	rd_link_ecc_uncorr_intr_clr	R/W1C	Clear Read Link-ECC uncorrectable error interrupt. If this bit set, rd_linkecc_uncorr_err_intr will be cleared. LPDDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Testable:** readOnly **Programming Mode:** Dynamic
4	rd_link_ecc_uncorr_intr_en	R/W	Interrupt enable bit for Read Link-ECC uncorrectable error. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic
3	rd_link_ecc_corr_intr_force	R/W1C	Interrupt force bit for rd_linkecc_corr_err_intr. Setting this register will cause the output interrupt to be asserted. DDRCTL automatically clears this bit. There is no interaction between functionally triggering an interrupt and forcing an interrupt (they are mutually exclusive). **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Testable:** readOnly **Programming Mode:** Dynamic
2	rd_link_ecc_corr_cnt_clr	R/W1C	Clear all Read Link-ECC correctable error count. If this bit set,LNKECCERRCNT0.rd_link_ecc_corr_cnt will be cleared. LPDDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Testable:** readOnly **Programming Mode:** Dynamic
1	rd_link_ecc_corr_intr_clr	R/W1C	Clear Read Link-ECC correctable error interrupt. If this bit set, rd_linkecc_corr_err_intr will be cleared. LPDDRCTL automatically clears this bit. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
0	rd_link_ecc_corr_intr_en	R/W	Interrupt enable bit for Read Link-ECC correctable error. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic

1.3.46 LNKECCPOISONCTL0

- **Description:** Link-ECC Poison Control Register 0
- **Size:** 32 bits
- **Offset:** 0x11988
- **Exists:** MEMC_LINK_ECC==1

This register is in block REGB_DDRC_CH1.

x:24	x:16	15:3	2	1	0
linkecc_poison_byte_sel	linkecc_poison_dmi_sel	Rsvd	linkecc_poison_rw	linkecc_poison_type	linkecc_poison_inject_en

Table 1-227 Fields for Register: LNKECCPOISONCTL0

Bits	Name	Memory Access	Description
x:24	linkecc_poison_byte_sel	R/W	Select target byte(s) of Data for Read/Write Link ECC poisoning. This is bit map indicator. Bit N corresponding to Data[N*8+:8]. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic **RangeVariable[x]:**"MEMC_DRAM_TOTAL_DATA_WIDTH/8" + 23
x:16	linkecc_poison_dmi_sel	R/W	Select target DMI(s) of Data for Write Link ECC poisoning. This is bit map indicator. Bit N corresponding to DMI[N]. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic **RangeVariable[x]:**"MEMC_DRAM_TOTAL_DATA_WIDTH/8" + 15
15:3			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
2	linkecc_poison_rw	R/W	Indicates whether the Link-ECC poisoning operation is Read or Write. ■ 0 - Write ■ 1 - Read **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic
1	linkecc_poison_type	R/W	Indicates whether the Link-ECC poisoning operation is Single-bit error or Double bit error. ■ 0 - Single bit Error ■ 1 - Double bit Error **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic
0	linkecc_poison_inject_en	R/W	Setting this register bit to 1 triggers the Link-ECC poisoning. Once Link-ECC is poisoned to a ECC code, the ECC poisoning is completed automatically and LNKECCPOISONSTAT.linkecc_poison_complete becomes 1. Please make sure that LNKECCPOISONSTAT.linkecc_poison_complete==0 before writing this register to 1. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic

1.3.47 LNKECCPOISONSTAT

- **Description:** Link-ECC Poison Status Register
- **Size:** 32 bits
- **Offset:** 0x1198c
- **Exists:** MEMC_LINK_ECC==1

This register is in block REGB_DDRC_CH1.

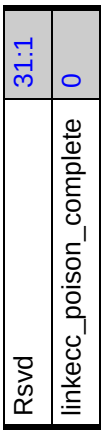


Table 1-228 Fields for Register: LNKECCPOISONSTAT

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	linkecc_poison_complete	R	Indicates Link-ECC poisoning operation is done. ■ 0 - Link-ECC poisoning is not completed ■ 1 - Link-ECC poisoning is completed **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic

1.3.48 LNKECCINDEX

- **Description:** Link-ECC Index Register
- **Size:** 32 bits
- **Offset:** 0x11990
- **Exists:** MEMC_LINK_ECC==1

This register is in block REGB_DDRC_CH1.

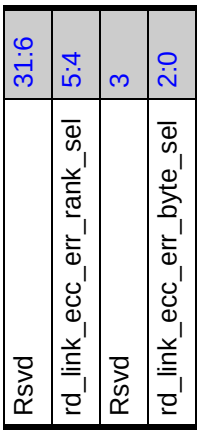


Table 1-229 Fields for Register: LNKECCINDEX

Bits	Name	Memory Access	Description
31:6			**Reserved Field:** Yes
5:4	rd_link_ecc_err_rank_sel	R/W	Select of which rank status output to LNKECCERRCNT.rd_link_ecc_uncorr_cnt, rd_link_ecc_corr_cnt and rd_link_ecc_err_syndrome. The value must be less than MEMC_NUM_RANKS. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 && MEMC_NUM_RANKS_GT_1==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1
3			**Reserved Field:** Yes
2:0	rd_link_ecc_err_byte_sel	R/W	Select of which data byte status output to LNKECCERRCNT.rd_link_ecc_uncorr_cnt, rd_link_ecc_corr_cnt and rd_link_ecc_err_syndrome. The value must be less than MEMC_DRAM_DATA_WIDTH/8. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1

1.3.49 LNKECCERRCNT0

- **Description:** Link-ECC Error Status Register 0
- **Size:** 32 bits
- **Offset:** 0x11994
- **Exists:** MEMC_LINK_ECC==1

This register is in block REGB_DDRC_CH1.

31:24	23:16	15:9	8:0
rd_link_ecc_uncorr_cnt	rd_link_ecc_corr_cnt	Rsvd	rd_link_ecc_err_syndrome

Table 1-230 Fields for Register: LNKECCERRCNT0

Bits	Name	Memory Access	Description
31:24	rd_link_ecc_uncorr_cnt	R	Indicates double bit error count. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic
23:16	rd_link_ecc_corr_cnt	R	Indicates single bit error count. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic
15:9			**Reserved Field:** Yes
8:0	rd_link_ecc_err_syndrome	R	Indicates ECC syndrome from most recent single bit error. **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Dynamic

1.3.50 LNKECCERRSTAT

- **Description:** Link-ECC Error Status Register 1
- **Size:** 32 bits
- **Offset:** 0x11998
- **Exists:** MEMC_LINK_ECC==1

This register is in block REGB_DDRC_CH1.

31:12	11:8	7:4	3:0
Rsvd	rd_link_ecc_uncorr_err_int	Rsvd	rd_link_ecc_corr_err_int

Table 1-231 Fields for Register: LNKECCERRSTAT

Bits	Name	Memory Access	Description
31:12			**Reserved Field:** Yes
11:8	rd_link_ecc_uncorr_err_int	R	Indicates double bit error for Read Link-ECC. If double bit error happens, this interrupt bit is set. It remains set until cleared by LNKECCCTL1.rd_link_ecc_uncorr_intr_clr. Each bit represents one rank. (LSB is the lowest rank number.) **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Static
7:4			**Reserved Field:** Yes
3:0	rd_link_ecc_corr_err_int	R	Indicates single bit error for Read Link-ECC. If signle bit error happens, this interrupt bit is set. It remains set until cleared by LNKECCCTL1.rd_link_ecc_corr_intr_clr. Each bit represents one rank. (LSB is the lowest rank number.) **Value After Reset:** 0x0 **Exists:** MEMC_LINK_ECC==1 **Programming Mode:** Static

1.3.51 OPCTRL1

- **Description:** Operation Control Register 1
- **Size:** 32 bits
- **Offset:** 0x11b84
- **Exists:** Always

This register is in block REGB_DDRC_CH1.

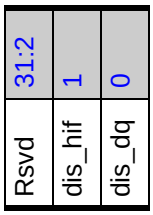


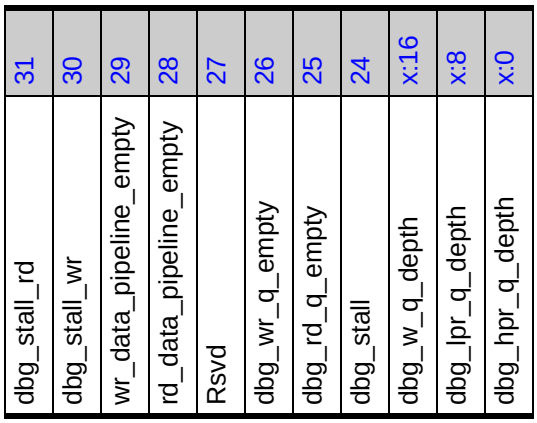
Table 1-232 Fields for Register: OPCTRL1

Bits	Name	Memory Access	Description
31:2			**Reserved Field:** Yes
1	dis_hif	R/W	When 1, DDRCTL asserts the HIF command signal hif_cmd_stall. DDRCTL will ignore the hif_cmd_valid and all other associated request signals. This bit is intended to be switched on-the-fly. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
0	dis_dq	R/W	When 1, DDRCTL will not de-queue any transactions from the CAM. Bypass is also disabled. All transactions are queued in the CAM. No reads or writes are issued to SDRAM as long as this is asserted. This bit may be used to prevent reads or writes being issued by the DDRCTL, which makes it safe to modify certain register fields associated with reads and writes (see Programming Chapter for details). After setting this bit, it is strongly recommended to poll OPCTRLCAM.wr_data_pipeline_empty and OPCTRLCAM.rd_data_pipeline_empty, before making changes to any registers which affect reads and writes. This will ensure that the relevant logic in the DDRC is idle. This bit is intended to be switched on-the-fly. Note: This bit is not applicable for designs working in DDR5 mode. In DDR5 mode, use software command interface command DisDqRef to achieve the same function as this bit. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.3.52 OPCTRLCAM

- **Description:** CAM Operation Control Register
- **Size:** 32 bits
- **Offset:** 0x11b88
- **Exists:** Always

This register is in block REGB_DDRC_CH1.



31	30	29	28	27	26	25	24	x:16	x:8	x:0
dbg_stall_rd	dbg_stall_wr	wr_data_pipeline_empty	rd_data_pipeline_empty	Rsvd	dbg_wr_q_empty	dbg_rd_q_empty	dbg_stall	dbg_w_q_depth	dbg_lpr_q_depth	dbg_hpr_q_depth

Table 1-233 Fields for Register: OPCTRLCAM

Bits	Name	Memory Access	Description
31	dbg_stall_rd	R	Stall for Read channel FOR DEBUG ONLY **Value After Reset:** 0x0 **Exists:** UMCTL2_DUAL_HIF_1==1 **Programming Mode:** Dynamic
30	dbg_stall_wr	R	Stall for Write channel FOR DEBUG ONLY **Value After Reset:** 0x0 **Exists:** UMCTL2_DUAL_HIF_1==1 **Programming Mode:** Dynamic
29	wr_data_pipeline_empty	R	This bit indicates that the write data pipeline on the DFI interface is empty. This register is intended to be polled at least twice after setting OPCTRL1.dis_dq, to ensure that all remaining commands/data have completed. **Value After Reset:** 0x0 **Exists:** Always **Reset Mask:** 0x0 **Volatile:** true **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
28	rd_data_pipeline_empty	R	This bit indicates that the read data pipeline on the DFI interface is empty. This register is intended to be polled at least twice after setting OPCTRL1.dis_dq, to ensure that all remaining commands/data have completed. **Value After Reset:** 0x0 **Exists:** Always **Reset Mask:** 0x0 **Volatile:** true **Programming Mode:** Dynamic
27			**Reserved Field:** Yes
26	dbg_wr_q_empty	R	When 1, all the Write command queues and Write data buffers inside DDRC are empty. This register is to be used for debug purpose. An example use-case scenario: When Controller enters Self-Refresh using the Low-Power entry sequence, Controller is expected to have executed all the commands in its queues and the write and read data drained. Hence this register must be 1 at that time. **Value After Reset:** 0x0 **Exists:** Always **Reset Mask:** 0x0 **Volatile:** true **Programming Mode:** Dynamic
25	dbg_rd_q_empty	R	When 1, all the Read command queues and Read data buffers inside DDRC are empty. This register is to be used for debug purpose. An example use-case scenario: When Controller enters Self-Refresh using the Low-Power entry sequence, Controller is expected to have executed all the commands in its queues and the write and read data drained. Hence this register must be 1 at that time. **Value After Reset:** 0x0 **Exists:** Always **Reset Mask:** 0x0 **Volatile:** true **Programming Mode:** Dynamic
24	dbg_stall	R	Stall FOR DEBUG ONLY **Value After Reset:** 0x0 **Exists:** UMCTL2_DUAL_HIF_1==0 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
x:16	dbg_w_q_depth	R	Write queue depth The last entry of WR queue is reserved for ECC SCRUB operation. This entry is not included in the calculation of the queue depth. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_WRCMD_ENTRY_BITS + 1" + 15
x:8	dbg_lpr_q_depth	R	Low priority read queue depth The last entry of Lpr queue is reserved for ECC SCRUB operation. This entry is not included in the calculation of the queue depth. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_RDCMD_ENTRY_BITS + 1" + 7
x:0	dbg_hpr_q_depth	R	High priority read queue depth **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_RDCMD_ENTRY_BITS + 1" - 1

1.3.53 OPCTRLCMD

- **Description:** Command Operation Control Register
- **Size:** 32 bits
- **Offset:** 0x11b8c
- **Exists:** Always

This register is in block REGB_DDRC_CH1.

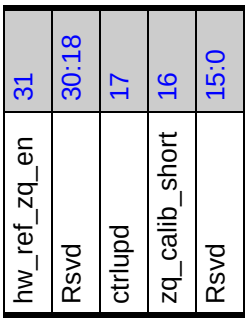


Table 1-234 Fields for Register: OPCTRLCMD

Bits	Name	Memory Access	Description
31	hw_ref_zq_en	R/W	Setting this register bit to 1 allows refresh and ZQCS/MPC(ZQ Calibration) commands to be triggered from hardware via the IOs ext_*. If set to 1, the fields OPCTRLCMD.zq_calib_short and OPREFCTRL*.rank*_refresh have no function, and are ignored by the DDRCTL logic. Setting this register bit to 0 allows refresh and ZQCS/MPC(ZQ Calibration) to be triggered from software, via the fields OPCTRLCMD.zq_calib_short and OPREFCTRL*.rank*_refresh. If set to 0, the hardware pins ext_* have no function, and are ignored by the DDRCTL logic. This register is static, and may only be changed when the DDRC reset signal, core_ddrc_rstn, is asserted (0). Note: Supporting this register field in this release is limited. Contact Synopsys if you wish to use this. Note: This field is not applicable for DDR5 ZQ Calibration. **Value After Reset:** 0x0 **Exists:** UMCTL2_REF_ZQ_IO==1 **Programming Mode:** Static
30:18			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
17	ctrlupd	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a dfi_ctrlupd_req to the PHY. When this request is stored in the DDRCTL, the bit is automatically cleared. This operation must only be performed when DFIUPD0.dis_auto_ctrlupd=1. Note: This field is not applicable for DDR5. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
16	zq_calib_short	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a ZQCS (ZQ calibration short)/MPC(ZQ calibration) command to the SDRAM. When this request is stored in the DDRCTL, the bit is automatically cleared. This operation can be performed only when ZQCTL0.dis_auto_zq=1. It is recommended NOT to set this register bit if in Init, in Self-Refresh(except LPDDR4/5) or SR-Powerdown(LPDDR4/5) or Deep Sleep Mode or Maximum Power Saving Mode. For Self-Refresh(except LPDDR4/5) or SR-Powerdown(LPDDR4/5) it will be scheduled after SR(except LPDDR4/5) or SRPD(LPDDR4/5) has been exited. For Deep Sleep Mode, it will be scheduled after DSM and/or SRPD has been exited. For Maximum Power Saving Mode, it will not be scheduled, although OPCTRLSTAT.zq_calib_short_busy will be de-asserted. Note: This field is not applicable for DDR5. **Value After Reset:** 0x0 **Exists:** Always **Testable:** readOnly **Programming Mode:** Dynamic
15:0			**Reserved Field:** Yes

1.3.54 OPCTRLSTAT

- **Description:** Status Operation Control Register
- **Size:** 32 bits
- **Offset:** 0x11b90
- **Exists:** Always

This register is in block REGB_DDRC_CH1.

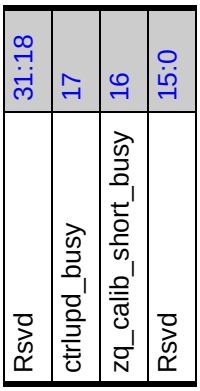


Table 1-235 Fields for Register: OPCTRLSTAT

Bits	Name	Memory Access	Description
31:18			**Reserved Field:** Yes
17	ctrlupd_busy	R	SoC core may initiate a ctrlupd operation only if this signal is low. This signal goes high in the clock after the DDRCTL accepts the ctrlupd request. It goes low when the ctrlupd operation is initiated in the DDRCTL. It is recommended not to perform ctrlupd operations when this signal is high. ■ 0 - Indicates that the SoC core can initiate a ctrlupd operation ■ 1 - Indicates that ctrlupd operation has not been initiated yet in the DDRCTL Note: This field is not applicable for DDR5. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
16	zq_calib_short_busy	R	SoC core may initiate a ZQCS (ZQ calibration short) operation only if this signal is low. This signal goes high in the clock after the DDRCTL accepts the ZQCS request. It goes low when the ZQCS operation is initiated in the DDRCTL. It is recommended not to perform ZQCS operations when this signal is high. ■ 0 - Indicates that the SoC core can initiate a ZQCS operation ■ 1 - Indicates that ZQCS operation has not been initiated yet in the DDRCTL Note: This field is not applicable for DDR5. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
15:0			**Reserved Field:** Yes

1.3.55 OPCTRLCAM1

- **Description:** CAM Operation Control Register 1
- **Size:** 32 bits
- **Offset:** 0x11b94
- **Exists:** MEMC_INLINE_ECC==1

This register is in block REGB_DDRC_CH1.

Table 1-236 Fields for Register: OPCTRLCAM1

Bits	Name	Memory Access	Description
x:0	dbg_wrecc_q_depth	R	Write ECC queue depth FOR DEBUG ONLY **Value After Reset:** 0x0 **Exists:** MEMC_INLINE_ECC==1 **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_WRCMD_ENTRY_BITS + 1" - 1

1.3.56 OPREFCTRL0

- **Description:** Refresh Operation Control Register 0
- **Size:** 32 bits
- **Offset:** 0x11b98
- **Exists:** Always

This register is in block REGB_DDRC_CH1.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
rank31_refresh	rank30_refresh	rank29_refresh	rank28_refresh	rank27_refresh	rank26_refresh	rank25_refresh	rank24_refresh	rank23_refresh	rank22_refresh	rank21_refresh	rank20_refresh	rank19_refresh	rank18_refresh	rank17_refresh	rank16_refresh	rank15_refresh	rank14_refresh	rank13_refresh	rank12_refresh	rank11_refresh	rank10_refresh	rank9_refresh	rank8_refresh	rank7_refresh	rank6_refresh	rank5_refresh	rank4_refresh	rank3_refresh	rank2_refresh	rank1_refresh	rank0_refresh

Table 1-237 Fields for Register: OPREFCTRL0

Bits	Name	Memory Access	Description
31	rank31_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 31. Writing to this bit causes OPREFSTAT0.rank31_refresh_busy to be set. When OPREFSTAT0.rank31_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 31. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>31 **Testable:** readOnly **Programming Mode:** Dynamic
30	rank30_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 30. Writing to this bit causes OPREFSTAT0.rank30_refresh_busy to be set. When OPREFSTAT0.rank30_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 30. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>30 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
29	rank29_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 29. Writing to this bit causes OPREFSTAT0.rank29_refresh_busy to be set. When OPREFSTAT0.rank29_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 29. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>29 **Testable:** readOnly **Programming Mode:** Dynamic
28	rank28_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 28. Writing to this bit causes OPREFSTAT0.rank28_refresh_busy to be set. When OPREFSTAT0.rank28_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 28. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>28 **Testable:** readOnly **Programming Mode:** Dynamic
27	rank27_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 27. Writing to this bit causes OPREFSTAT0.rank27_refresh_busy to be set. When OPREFSTAT0.rank27_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 27. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>27 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
26	rank26_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 26. Writing to this bit causes OPREFSTAT0.rank26_refresh_busy to be set. When OPREFSTAT0.rank26_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 26. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>26 **Testable:** readOnly **Programming Mode:** Dynamic
25	rank25_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 25. Writing to this bit causes OPREFSTAT0.rank25_refresh_busy to be set. When OPREFSTAT0.rank25_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 25. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>25 **Testable:** readOnly **Programming Mode:** Dynamic
24	rank24_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 24. Writing to this bit causes OPREFSTAT0.rank24_refresh_busy to be set. When OPREFSTAT0.rank24_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 24. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>24 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
23	rank23_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 23. Writing to this bit causes OPREFSTAT0.rank23_refresh_busy to be set. When OPREFSTAT0.rank23_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 23. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>23 **Testable:** readOnly **Programming Mode:** Dynamic
22	rank22_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 22. Writing to this bit causes OPREFSTAT0.rank22_refresh_busy to be set. When OPREFSTAT0.rank22_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 22. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>22 **Testable:** readOnly **Programming Mode:** Dynamic
21	rank21_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 21. Writing to this bit causes OPREFSTAT0.rank21_refresh_busy to be set. When OPREFSTAT0.rank21_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 21. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>21 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
20	rank20_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 20. Writing to this bit causes OPREFSTAT0.rank20_refresh_busy to be set. When OPREFSTAT0.rank20_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 20. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>20 **Testable:** readOnly **Programming Mode:** Dynamic
19	rank19_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 19. Writing to this bit causes OPREFSTAT0.rank19_refresh_busy to be set. When OPREFSTAT0.rank19_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 19. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>19 **Testable:** readOnly **Programming Mode:** Dynamic
18	rank18_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 18. Writing to this bit causes OPREFSTAT0.rank18_refresh_busy to be set. When OPREFSTAT0.rank18_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 18. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>18 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
17	rank17_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 17. Writing to this bit causes OPREFSTAT0.rank17_refresh_busy to be set. When OPREFSTAT0.rank17_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 17. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>17 **Testable:** readOnly **Programming Mode:** Dynamic
16	rank16_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 16. Writing to this bit causes OPREFSTAT0.rank16_refresh_busy to be set. When OPREFSTAT0.rank16_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 16. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>16 **Testable:** readOnly **Programming Mode:** Dynamic
15	rank15_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 15. Writing to this bit causes OPREFSTAT0.rank15_refresh_busy to be set. When OPREFSTAT0.rank15_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 15. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>15 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
14	rank14_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 14. Writing to this bit causes OPREFSTAT0.rank14_refresh_busy to be set. When OPREFSTAT0.rank14_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 14. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>14 **Testable:** readOnly **Programming Mode:** Dynamic
13	rank13_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 13. Writing to this bit causes OPREFSTAT0.rank13_refresh_busy to be set. When OPREFSTAT0.rank13_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 13. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>13 **Testable:** readOnly **Programming Mode:** Dynamic
12	rank12_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 12. Writing to this bit causes OPREFSTAT0.rank12_refresh_busy to be set. When OPREFSTAT0.rank12_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 12. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>12 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
11	rank11_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 11. Writing to this bit causes OPREFSTAT0.rank11_refresh_busy to be set. When OPREFSTAT0.rank11_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 11. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>11 **Testable:** readOnly **Programming Mode:** Dynamic
10	rank10_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 10. Writing to this bit causes OPREFSTAT0.rank10_refresh_busy to be set. When OPREFSTAT0.rank10_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 10. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>10 **Testable:** readOnly **Programming Mode:** Dynamic
9	rank9_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 9. Writing to this bit causes OPREFSTAT0.rank9_refresh_busy to be set. When OPREFSTAT0.rank9_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 9. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>9 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
8	rank8_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 8. Writing to this bit causes OPREFSTAT0.rank8_refresh_busy to be set. When OPREFSTAT0.rank8_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 8. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>8 **Testable:** readOnly **Programming Mode:** Dynamic
7	rank7_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 7. Writing to this bit causes OPREFSTAT0.rank7_refresh_busy to be set. When OPREFSTAT0.rank7_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 7. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>7 **Testable:** readOnly **Programming Mode:** Dynamic
6	rank6_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 6. Writing to this bit causes OPREFSTAT0.rank6_refresh_busy to be set. When OPREFSTAT0.rank6_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 6. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>6 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
5	rank5_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 5. Writing to this bit causes OPREFSTAT0.rank5_refresh_busy to be set. When OPREFSTAT0.rank5_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 5. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>5 **Testable:** readOnly **Programming Mode:** Dynamic
4	rank4_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 4. Writing to this bit causes OPREFSTAT0.rank4_refresh_busy to be set. When OPREFSTAT0.rank4_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 4. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>4 **Testable:** readOnly **Programming Mode:** Dynamic
3	rank3_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 3. Writing to this bit causes OPREFSTAT0.rank3_refresh_busy to be set. When OPREFSTAT0.rank3_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 3. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>3 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
2	rank2_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 2. Writing to this bit causes OPREFSTAT0.rank2_refresh_busy to be set. When OPREFSTAT0.rank2_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 2. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>2 **Testable:** readOnly **Programming Mode:** Dynamic
1	rank1_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 1. Writing to this bit causes OPREFSTAT0.rank1_refresh_busy to be set. When OPREFSTAT0.rank1_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 1. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>1 **Testable:** readOnly **Programming Mode:** Dynamic
0	rank0_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 0. Writing to this bit causes OPREFSTAT0.rank0_refresh_busy to be set. When OPREFSTAT0.rank0_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 0. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>0 **Testable:** readOnly **Programming Mode:** Dynamic

1.3.57 OPREFCTRL1

- **Description:** Refresh Operation Control Register 1
- **Size:** 32 bits
- **Offset:** 0x11b9c
- **Exists:** UMCTL2_NUM_LRANKS_TOTAL>32

This register is in block REGB_DDRC_CH1.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
rank63_refresh	rank62_refresh	rank61_refresh	rank60_refresh	rank59_refresh	rank58_refresh	rank57_refresh	rank56_refresh	rank55_refresh	rank54_refresh	rank53_refresh	rank52_refresh	rank51_refresh	rank50_refresh	rank49_refresh	rank48_refresh	rank47_refresh	rank46_refresh	rank45_refresh	rank44_refresh	rank43_refresh	rank42_refresh	rank41_refresh	rank40_refresh	rank39_refresh	rank38_refresh	rank37_refresh	rank36_refresh	rank35_refresh	rank34_refresh	rank33_refresh	rank32_refresh

Table 1-238 Fields for Register: OPREFCTRL1

Bits	Name	Memory Access	Description
31	rank63_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 63. Writing to this bit causes OPREFSTAT1.rank63_refresh_busy to be set. When OPREFSTAT1.rank63_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 63. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>63 **Testable:** readOnly **Programming Mode:** Dynamic
30	rank62_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 62. Writing to this bit causes OPREFSTAT1.rank62_refresh_busy to be set. When OPREFSTAT1.rank62_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 62. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>62 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
29	rank61_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 61. Writing to this bit causes OPREFSTAT1.rank61_refresh_busy to be set. When OPREFSTAT1.rank61_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 61. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>61 **Testable:** readOnly **Programming Mode:** Dynamic
28	rank60_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 60. Writing to this bit causes OPREFSTAT1.rank60_refresh_busy to be set. When OPREFSTAT1.rank60_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 60. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>60 **Testable:** readOnly **Programming Mode:** Dynamic
27	rank59_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 59. Writing to this bit causes OPREFSTAT1.rank59_refresh_busy to be set. When OPREFSTAT1.rank59_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 59. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>59 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
26	rank58_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 58. Writing to this bit causes OPREFSTAT1.rank58_refresh_busy to be set. When OPREFSTAT1.rank58_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 58. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>58 **Testable:** readOnly **Programming Mode:** Dynamic
25	rank57_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 57. Writing to this bit causes OPREFSTAT1.rank57_refresh_busy to be set. When OPREFSTAT1.rank57_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 57. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>57 **Testable:** readOnly **Programming Mode:** Dynamic
24	rank56_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 56. Writing to this bit causes OPREFSTAT1.rank56_refresh_busy to be set. When OPREFSTAT1.rank56_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 56. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>56 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
23	rank55_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 55. Writing to this bit causes OPREFSTAT1.rank55_refresh_busy to be set. When OPREFSTAT1.rank55_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 55. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>55 **Testable:** readOnly **Programming Mode:** Dynamic
22	rank54_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 54. Writing to this bit causes OPREFSTAT1.rank54_refresh_busy to be set. When OPREFSTAT1.rank54_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 54. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>54 **Testable:** readOnly **Programming Mode:** Dynamic
21	rank53_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 53. Writing to this bit causes OPREFSTAT1.rank53_refresh_busy to be set. When OPREFSTAT1.rank53_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 53. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>53 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
20	rank52_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 52. Writing to this bit causes OPREFSTAT1.rank52_refresh_busy to be set. When OPREFSTAT1.rank52_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 52. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>52 **Testable:** readOnly **Programming Mode:** Dynamic
19	rank51_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 51. Writing to this bit causes OPREFSTAT1.rank51_refresh_busy to be set. When OPREFSTAT1.rank51_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 51. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>51 **Testable:** readOnly **Programming Mode:** Dynamic
18	rank50_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 50. Writing to this bit causes OPREFSTAT1.rank50_refresh_busy to be set. When OPREFSTAT1.rank50_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 50. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>50 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
17	rank49_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 49. Writing to this bit causes OPREFSTAT1.rank49_refresh_busy to be set. When OPREFSTAT1.rank49_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 49. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>49 **Testable:** readOnly **Programming Mode:** Dynamic
16	rank48_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 48. Writing to this bit causes OPREFSTAT1.rank48_refresh_busy to be set. When OPREFSTAT1.rank48_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 48. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>48 **Testable:** readOnly **Programming Mode:** Dynamic
15	rank47_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 47. Writing to this bit causes OPREFSTAT1.rank47_refresh_busy to be set. When OPREFSTAT1.rank47_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 47. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>47 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
14	rank46_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 46. Writing to this bit causes OPREFSTAT1.rank46_refresh_busy to be set. When OPREFSTAT1.rank46_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 46. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>46 **Testable:** readOnly **Programming Mode:** Dynamic
13	rank45_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 45. Writing to this bit causes OPREFSTAT1.rank45_refresh_busy to be set. When OPREFSTAT1.rank45_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 45. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>45 **Testable:** readOnly **Programming Mode:** Dynamic
12	rank44_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 44. Writing to this bit causes OPREFSTAT1.rank44_refresh_busy to be set. When OPREFSTAT1.rank44_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 44. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>44 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
11	rank43_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 43. Writing to this bit causes OPREFSTAT1.rank43_refresh_busy to be set. When OPREFSTAT1.rank43_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 43. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>43 **Testable:** readOnly **Programming Mode:** Dynamic
10	rank42_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 42. Writing to this bit causes OPREFSTAT1.rank42_refresh_busy to be set. When OPREFSTAT1.rank42_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 42. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>42 **Testable:** readOnly **Programming Mode:** Dynamic
9	rank41_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 41. Writing to this bit causes OPREFSTAT1.rank41_refresh_busy to be set. When OPREFSTAT1.rank41_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 41. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>41 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
8	rank40_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 40. Writing to this bit causes OPREFSTAT1.rank40_refresh_busy to be set. When OPREFSTAT1.rank40_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 40. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>40 **Testable:** readOnly **Programming Mode:** Dynamic
7	rank39_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 39. Writing to this bit causes OPREFSTAT1.rank39_refresh_busy to be set. When OPREFSTAT1.rank39_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 39. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>39 **Testable:** readOnly **Programming Mode:** Dynamic
6	rank38_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 38. Writing to this bit causes OPREFSTAT1.rank38_refresh_busy to be set. When OPREFSTAT1.rank38_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 38. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>38 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
5	rank37_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 37. Writing to this bit causes OPREFSTAT1.rank37_refresh_busy to be set. When OPREFSTAT1.rank37_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 37. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>37 **Testable:** readOnly **Programming Mode:** Dynamic
4	rank36_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 36. Writing to this bit causes OPREFSTAT1.rank36_refresh_busy to be set. When OPREFSTAT1.rank36_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 36. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>36 **Testable:** readOnly **Programming Mode:** Dynamic
3	rank35_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 35. Writing to this bit causes OPREFSTAT1.rank35_refresh_busy to be set. When OPREFSTAT1.rank35_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 35. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>35 **Testable:** readOnly **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
2	rank34_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 34. Writing to this bit causes OPREFSTAT1.rank34_refresh_busy to be set. When OPREFSTAT1.rank34_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 34. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>34 **Testable:** readOnly **Programming Mode:** Dynamic
1	rank33_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 33. Writing to this bit causes OPREFSTAT1.rank33_refresh_busy to be set. When OPREFSTAT1.rank33_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 33. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>33 **Testable:** readOnly **Programming Mode:** Dynamic
0	rank32_refresh	R/W1S	Setting this register bit to 1 indicates to the DDRCTL to issue a refresh to rank 32. Writing to this bit causes OPREFSTAT1.rank32_refresh_busy to be set. When OPREFSTAT1.rank32_refresh_busy is cleared, the command has been stored in the DDRCTL. For 3DS configuration, refresh is sent to logical rank index 32. This operation can be performed only when RFSHCTL0.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Maximum Power Saving Mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>32 **Testable:** readOnly **Programming Mode:** Dynamic

1.3.58 OPREFSTAT0

- **Description:** Refresh Operation Status Register 0
- **Size:** 32 bits
- **Offset:** 0x11ba0
- **Exists:** Always

This register is in block REGB_DDRC_CH1.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
rank31_refresh_busy	rank30_refresh_busy	rank29_refresh_busy	rank28_refresh_busy	rank27_refresh_busy	rank26_refresh_busy	rank25_refresh_busy	rank24_refresh_busy	rank23_refresh_busy	rank22_refresh_busy	rank21_refresh_busy	rank20_refresh_busy	rank19_refresh_busy	rank18_refresh_busy	rank17_refresh_busy	rank16_refresh_busy	rank15_refresh_busy	rank14_refresh_busy	rank13_refresh_busy	rank12_refresh_busy	rank11_refresh_busy	rank10_refresh_busy	rank9_refresh_busy	rank8_refresh_busy	rank7_refresh_busy	rank6_refresh_busy	rank5_refresh_busy	rank4_refresh_busy	rank3_refresh_busy	rank2_refresh_busy	rank1_refresh_busy	rank0_refresh_busy

Table 1-239 Fields for Register: OPREFSTAT0

Bits	Name	Memory Access	Description
31	rank31_refresh_busy	R	SoC core may initiate a rank31_refresh operation (refresh operation to rank 31) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank31_refresh is set to one. It goes low when the rank31_refresh operation is stored in the DDRCTL. It is recommended not to perform rank31_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank31_refresh operation ■ 1 - Indicates that rank31_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>31 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
30	rank30_refresh_busy	R	SoC core may initiate a rank30_refresh operation (refresh operation to rank 30) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank30_refresh is set to one. It goes low when the rank30_refresh operation is stored in the DDRCTL. It is recommended not to perform rank30_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank30_refresh operation ■ 1 - Indicates that rank30_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>30 **Programming Mode:** Dynamic
29	rank29_refresh_busy	R	SoC core may initiate a rank29_refresh operation (refresh operation to rank 29) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank29_refresh is set to one. It goes low when the rank29_refresh operation is stored in the DDRCTL. It is recommended not to perform rank29_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank29_refresh operation ■ 1 - Indicates that rank29_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>29 **Programming Mode:** Dynamic
28	rank28_refresh_busy	R	SoC core may initiate a rank28_refresh operation (refresh operation to rank 28) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank28_refresh is set to one. It goes low when the rank28_refresh operation is stored in the DDRCTL. It is recommended not to perform rank28_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank28_refresh operation ■ 1 - Indicates that rank28_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>28 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
27	rank27_refresh_busy	R	SoC core may initiate a rank27_refresh operation (refresh operation to rank 27) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank27_refresh is set to one. It goes low when the rank27_refresh operation is stored in the DDRCTL. It is recommended not to perform rank27_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank27_refresh operation ■ 1 - Indicates that rank27_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>27 **Programming Mode:** Dynamic
26	rank26_refresh_busy	R	SoC core may initiate a rank26_refresh operation (refresh operation to rank 26) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank26_refresh is set to one. It goes low when the rank26_refresh operation is stored in the DDRCTL. It is recommended not to perform rank26_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank26_refresh operation ■ 1 - Indicates that rank26_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>26 **Programming Mode:** Dynamic
25	rank25_refresh_busy	R	SoC core may initiate a rank25_refresh operation (refresh operation to rank 25) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank25_refresh is set to one. It goes low when the rank25_refresh operation is stored in the DDRCTL. It is recommended not to perform rank25_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank25_refresh operation ■ 1 - Indicates that rank25_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>25 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
24	rank24_refresh_busy	R	SoC core may initiate a rank24_refresh operation (refresh operation to rank 24) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank24_refresh is set to one. It goes low when the rank24_refresh operation is stored in the DDRCTL. It is recommended not to perform rank24_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank24_refresh operation ■ 1 - Indicates that rank24_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>24 **Programming Mode:** Dynamic
23	rank23_refresh_busy	R	SoC core may initiate a rank23_refresh operation (refresh operation to rank 23) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank23_refresh is set to one. It goes low when the rank23_refresh operation is stored in the DDRCTL. It is recommended not to perform rank23_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank23_refresh operation ■ 1 - Indicates that rank23_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>23 **Programming Mode:** Dynamic
22	rank22_refresh_busy	R	SoC core may initiate a rank22_refresh operation (refresh operation to rank 22) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank22_refresh is set to one. It goes low when the rank22_refresh operation is stored in the DDRCTL. It is recommended not to perform rank22_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank22_refresh operation ■ 1 - Indicates that rank22_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>22 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
21	rank21_refresh_busy	R	SoC core may initiate a rank21_refresh operation (refresh operation to rank 21) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank21_refresh is set to one. It goes low when the rank21_refresh operation is stored in the DDRCTL. It is recommended not to perform rank21_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank21_refresh operation ■ 1 - Indicates that rank21_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>21 **Programming Mode:** Dynamic
20	rank20_refresh_busy	R	SoC core may initiate a rank20_refresh operation (refresh operation to rank 20) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank20_refresh is set to one. It goes low when the rank20_refresh operation is stored in the DDRCTL. It is recommended not to perform rank20_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank20_refresh operation ■ 1 - Indicates that rank20_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>20 **Programming Mode:** Dynamic
19	rank19_refresh_busy	R	SoC core may initiate a rank19_refresh operation (refresh operation to rank 19) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank19_refresh is set to one. It goes low when the rank19_refresh operation is stored in the DDRCTL. It is recommended not to perform rank19_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank19_refresh operation ■ 1 - Indicates that rank19_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>19 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
18	rank18_refresh_busy	R	SoC core may initiate a rank18_refresh operation (refresh operation to rank 18) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank18_refresh is set to one. It goes low when the rank18_refresh operation is stored in the DDRCTL. It is recommended not to perform rank18_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank18_refresh operation ■ 1 - Indicates that rank18_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>18 **Programming Mode:** Dynamic
17	rank17_refresh_busy	R	SoC core may initiate a rank17_refresh operation (refresh operation to rank 17) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank17_refresh is set to one. It goes low when the rank17_refresh operation is stored in the DDRCTL. It is recommended not to perform rank17_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank17_refresh operation ■ 1 - Indicates that rank17_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>17 **Programming Mode:** Dynamic
16	rank16_refresh_busy	R	SoC core may initiate a rank16_refresh operation (refresh operation to rank 16) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank16_refresh is set to one. It goes low when the rank16_refresh operation is stored in the DDRCTL. It is recommended not to perform rank16_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank16_refresh operation ■ 1 - Indicates that rank16_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>16 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
15	rank15_refresh_busy	R	SoC core may initiate a rank15_refresh operation (refresh operation to rank 15) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank15_refresh is set to one. It goes low when the rank15_refresh operation is stored in the DDRCTL. It is recommended not to perform rank15_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank15_refresh operation ■ 1 - Indicates that rank15_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>15 **Programming Mode:** Dynamic
14	rank14_refresh_busy	R	SoC core may initiate a rank14_refresh operation (refresh operation to rank 14) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank14_refresh is set to one. It goes low when the rank14_refresh operation is stored in the DDRCTL. It is recommended not to perform rank14_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank14_refresh operation ■ 1 - Indicates that rank14_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>14 **Programming Mode:** Dynamic
13	rank13_refresh_busy	R	SoC core may initiate a rank13_refresh operation (refresh operation to rank 13) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank13_refresh is set to one. It goes low when the rank13_refresh operation is stored in the DDRCTL. It is recommended not to perform rank13_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank13_refresh operation ■ 1 - Indicates that rank13_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>13 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
12	rank12_refresh_busy	R	SoC core may initiate a rank12_refresh operation (refresh operation to rank 12) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank12_refresh is set to one. It goes low when the rank12_refresh operation is stored in the DDRCTL. It is recommended not to perform rank12_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank12_refresh operation ■ 1 - Indicates that rank12_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>12 **Programming Mode:** Dynamic
11	rank11_refresh_busy	R	SoC core may initiate a rank11_refresh operation (refresh operation to rank 11) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank11_refresh is set to one. It goes low when the rank11_refresh operation is stored in the DDRCTL. It is recommended not to perform rank11_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank11_refresh operation ■ 1 - Indicates that rank11_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>11 **Programming Mode:** Dynamic
10	rank10_refresh_busy	R	SoC core may initiate a rank10_refresh operation (refresh operation to rank 10) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank10_refresh is set to one. It goes low when the rank10_refresh operation is stored in the DDRCTL. It is recommended not to perform rank10_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank10_refresh operation ■ 1 - Indicates that rank10_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>10 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
9	rank9_refresh_busy	R	SoC core may initiate a rank9_refresh operation (refresh operation to rank 9) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank9_refresh is set to one. It goes low when the rank9_refresh operation is stored in the DDRCTL. It is recommended not to perform rank9_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank9_refresh operation ■ 1 - Indicates that rank9_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>9 **Programming Mode:** Dynamic
8	rank8_refresh_busy	R	SoC core may initiate a rank8_refresh operation (refresh operation to rank 8) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank8_refresh is set to one. It goes low when the rank8_refresh operation is stored in the DDRCTL. It is recommended not to perform rank8_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank8_refresh operation ■ 1 - Indicates that rank8_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>8 **Programming Mode:** Dynamic
7	rank7_refresh_busy	R	SoC core may initiate a rank7_refresh operation (refresh operation to rank 7) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank7_refresh is set to one. It goes low when the rank7_refresh operation is stored in the DDRCTL. It is recommended not to perform rank7_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank7_refresh operation ■ 1 - Indicates that rank7_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>7 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
6	rank6_refresh_busy	R	SoC core may initiate a rank6_refresh operation (refresh operation to rank 6) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank6_refresh is set to one. It goes low when the rank6_refresh operation is stored in the DDRCTL. It is recommended not to perform rank6_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank6_refresh operation ■ 1 - Indicates that rank6_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>6 **Programming Mode:** Dynamic
5	rank5_refresh_busy	R	SoC core may initiate a rank5_refresh operation (refresh operation to rank 5) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank5_refresh is set to one. It goes low when the rank5_refresh operation is stored in the DDRCTL. It is recommended not to perform rank5_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank5_refresh operation ■ 1 - Indicates that rank5_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>5 **Programming Mode:** Dynamic
4	rank4_refresh_busy	R	SoC core may initiate a rank4_refresh operation (refresh operation to rank 4) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank4_refresh is set to one. It goes low when the rank4_refresh operation is stored in the DDRCTL. It is recommended not to perform rank4_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank4_refresh operation ■ 1 - Indicates that rank4_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>4 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
3	rank3_refresh_busy	R	SoC core may initiate a rank3_refresh operation (refresh operation to rank 3) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank3_refresh is set to one. It goes low when the rank3_refresh operation is stored in the DDRCTL. It is recommended not to perform rank3_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank3_refresh operation ■ 1 - Indicates that rank3_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>3 **Programming Mode:** Dynamic
2	rank2_refresh_busy	R	SoC core may initiate a rank2_refresh operation (refresh operation to rank 2) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank2_refresh is set to one. It goes low when the rank2_refresh operation is stored in the DDRCTL. It is recommended not to perform rank2_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank2_refresh operation ■ 1 - Indicates that rank2_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>2 **Programming Mode:** Dynamic
1	rank1_refresh_busy	R	SoC core may initiate a rank1_refresh operation (refresh operation to rank 1) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank1_refresh is set to one. It goes low when the rank1_refresh operation is stored in the DDRCTL. It is recommended not to perform rank1_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank1_refresh operation ■ 1 - Indicates that rank1_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
0	rank0_refresh_busy	R	SoC core may initiate a rank0_refresh operation (refresh operation to rank 0) only if this signal is low. This signal goes high in the clock after OPREFCTRL0.rank0_refresh is set to one. It goes low when the rank0_refresh operation is stored in the DDRCTL. It is recommended not to perform rank0_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank0_refresh operation ■ 1 - Indicates that rank0_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>0 **Programming Mode:** Dynamic

1.3.59 OPREFSTAT1

- **Description:** Refresh Operation Status Register 1
- **Size:** 32 bits
- **Offset:** 0x11ba4
- **Exists:** UMCTL2_NUM_LRANKS_TOTAL>32

This register is in block REGB_DDRC_CH1.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
rank63_refresh_busy	rank62_refresh_busy	rank61_refresh_busy	rank60_refresh_busy	rank59_refresh_busy	rank58_refresh_busy	rank57_refresh_busy	rank56_refresh_busy	rank55_refresh_busy	rank54_refresh_busy	rank53_refresh_busy	rank52_refresh_busy	rank51_refresh_busy	rank50_refresh_busy	rank49_refresh_busy	rank48_refresh_busy	rank47_refresh_busy	rank46_refresh_busy	rank45_refresh_busy	rank44_refresh_busy	rank43_refresh_busy	rank42_refresh_busy	rank41_refresh_busy	rank40_refresh_busy	rank39_refresh_busy	rank38_refresh_busy	rank37_refresh_busy	rank36_refresh_busy	rank35_refresh_busy	rank34_refresh_busy	rank33_refresh_busy	rank32_refresh_busy

Table 1-240 Fields for Register: OPREFSTAT1

Bits	Name	Memory Access	Description
31	rank63_refresh_busy	R	SoC core may initiate a rank63_refresh operation (refresh operation to rank 63) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank63_refresh is set to one. It goes low when the rank63_refresh operation is stored in the DDRCTL. It is recommended not to perform rank63_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank63_refresh operation ■ 1 - Indicates that rank63_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>63 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
30	rank62_refresh_busy	R	SoC core may initiate a rank62_refresh operation (refresh operation to rank 62) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank62_refresh is set to one. It goes low when the rank62_refresh operation is stored in the DDRCTL. It is recommended not to perform rank62_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank62_refresh operation ■ 1 - Indicates that rank62_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>62 **Programming Mode:** Dynamic
29	rank61_refresh_busy	R	SoC core may initiate a rank61_refresh operation (refresh operation to rank 61) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank61_refresh is set to one. It goes low when the rank61_refresh operation is stored in the DDRCTL. It is recommended not to perform rank61_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank61_refresh operation ■ 1 - Indicates that rank61_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>61 **Programming Mode:** Dynamic
28	rank60_refresh_busy	R	SoC core may initiate a rank60_refresh operation (refresh operation to rank 60) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank60_refresh is set to one. It goes low when the rank60_refresh operation is stored in the DDRCTL. It is recommended not to perform rank60_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank60_refresh operation ■ 1 - Indicates that rank60_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>60 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
27	rank59_refresh_busy	R	SoC core may initiate a rank59_refresh operation (refresh operation to rank 59) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank59_refresh is set to one. It goes low when the rank59_refresh operation is stored in the DDRCTL. It is recommended not to perform rank59_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank59_refresh operation ■ 1 - Indicates that rank59_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>59 **Programming Mode:** Dynamic
26	rank58_refresh_busy	R	SoC core may initiate a rank58_refresh operation (refresh operation to rank 58) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank58_refresh is set to one. It goes low when the rank58_refresh operation is stored in the DDRCTL. It is recommended not to perform rank58_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank58_refresh operation ■ 1 - Indicates that rank58_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>58 **Programming Mode:** Dynamic
25	rank57_refresh_busy	R	SoC core may initiate a rank57_refresh operation (refresh operation to rank 57) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank57_refresh is set to one. It goes low when the rank57_refresh operation is stored in the DDRCTL. It is recommended not to perform rank57_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank57_refresh operation ■ 1 - Indicates that rank57_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>57 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
24	rank56_refresh_busy	R	SoC core may initiate a rank56_refresh operation (refresh operation to rank 56) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank56_refresh is set to one. It goes low when the rank56_refresh operation is stored in the DDRCTL. It is recommended not to perform rank56_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank56_refresh operation ■ 1 - Indicates that rank56_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>56 **Programming Mode:** Dynamic
23	rank55_refresh_busy	R	SoC core may initiate a rank55_refresh operation (refresh operation to rank 55) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank55_refresh is set to one. It goes low when the rank55_refresh operation is stored in the DDRCTL. It is recommended not to perform rank55_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank55_refresh operation ■ 1 - Indicates that rank55_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>55 **Programming Mode:** Dynamic
22	rank54_refresh_busy	R	SoC core may initiate a rank54_refresh operation (refresh operation to rank 54) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank54_refresh is set to one. It goes low when the rank54_refresh operation is stored in the DDRCTL. It is recommended not to perform rank54_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank54_refresh operation ■ 1 - Indicates that rank54_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>54 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
21	rank53_refresh_busy	R	SoC core may initiate a rank53_refresh operation (refresh operation to rank 53) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank53_refresh is set to one. It goes low when the rank53_refresh operation is stored in the DDRCTL. It is recommended not to perform rank53_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank53_refresh operation ■ 1 - Indicates that rank53_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>53 **Programming Mode:** Dynamic
20	rank52_refresh_busy	R	SoC core may initiate a rank52_refresh operation (refresh operation to rank 52) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank52_refresh is set to one. It goes low when the rank52_refresh operation is stored in the DDRCTL. It is recommended not to perform rank52_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank52_refresh operation ■ 1 - Indicates that rank52_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>52 **Programming Mode:** Dynamic
19	rank51_refresh_busy	R	SoC core may initiate a rank51_refresh operation (refresh operation to rank 51) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank51_refresh is set to one. It goes low when the rank51_refresh operation is stored in the DDRCTL. It is recommended not to perform rank51_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank51_refresh operation ■ 1 - Indicates that rank51_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>51 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
18	rank50_refresh_busy	R	SoC core may initiate a rank50_refresh operation (refresh operation to rank 50) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank50_refresh is set to one. It goes low when the rank50_refresh operation is stored in the DDRCTL. It is recommended not to perform rank50_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank50_refresh operation ■ 1 - Indicates that rank50_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>50 **Programming Mode:** Dynamic
17	rank49_refresh_busy	R	SoC core may initiate a rank49_refresh operation (refresh operation to rank 49) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank49_refresh is set to one. It goes low when the rank49_refresh operation is stored in the DDRCTL. It is recommended not to perform rank49_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank49_refresh operation ■ 1 - Indicates that rank49_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>49 **Programming Mode:** Dynamic
16	rank48_refresh_busy	R	SoC core may initiate a rank48_refresh operation (refresh operation to rank 48) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank48_refresh is set to one. It goes low when the rank48_refresh operation is stored in the DDRCTL. It is recommended not to perform rank48_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank48_refresh operation ■ 1 - Indicates that rank48_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>48 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
15	rank47_refresh_busy	R	SoC core may initiate a rank47_refresh operation (refresh operation to rank 47) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank47_refresh is set to one. It goes low when the rank47_refresh operation is stored in the DDRCTL. It is recommended not to perform rank47_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank47_refresh operation ■ 1 - Indicates that rank47_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>47 **Programming Mode:** Dynamic
14	rank46_refresh_busy	R	SoC core may initiate a rank46_refresh operation (refresh operation to rank 46) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank46_refresh is set to one. It goes low when the rank46_refresh operation is stored in the DDRCTL. It is recommended not to perform rank46_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank46_refresh operation ■ 1 - Indicates that rank46_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>46 **Programming Mode:** Dynamic
13	rank45_refresh_busy	R	SoC core may initiate a rank45_refresh operation (refresh operation to rank 45) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank45_refresh is set to one. It goes low when the rank45_refresh operation is stored in the DDRCTL. It is recommended not to perform rank45_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank45_refresh operation ■ 1 - Indicates that rank45_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>45 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
12	rank44_refresh_busy	R	SoC core may initiate a rank44_refresh operation (refresh operation to rank 44) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank44_refresh is set to one. It goes low when the rank44_refresh operation is stored in the DDRCTL. It is recommended not to perform rank44_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank44_refresh operation ■ 1 - Indicates that rank44_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>44 **Programming Mode:** Dynamic
11	rank43_refresh_busy	R	SoC core may initiate a rank43_refresh operation (refresh operation to rank 43) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank43_refresh is set to one. It goes low when the rank43_refresh operation is stored in the DDRCTL. It is recommended not to perform rank43_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank43_refresh operation ■ 1 - Indicates that rank43_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>43 **Programming Mode:** Dynamic
10	rank42_refresh_busy	R	SoC core may initiate a rank42_refresh operation (refresh operation to rank 42) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank42_refresh is set to one. It goes low when the rank42_refresh operation is stored in the DDRCTL. It is recommended not to perform rank42_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank42_refresh operation ■ 1 - Indicates that rank42_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>42 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
9	rank41_refresh_busy	R	SoC core may initiate a rank41_refresh operation (refresh operation to rank 41) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank41_refresh is set to one. It goes low when the rank41_refresh operation is stored in the DDRCTL. It is recommended not to perform rank41_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank41_refresh operation ■ 1 - Indicates that rank41_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>41 **Programming Mode:** Dynamic
8	rank40_refresh_busy	R	SoC core may initiate a rank40_refresh operation (refresh operation to rank 40) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank40_refresh is set to one. It goes low when the rank40_refresh operation is stored in the DDRCTL. It is recommended not to perform rank40_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank40_refresh operation ■ 1 - Indicates that rank40_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>40 **Programming Mode:** Dynamic
7	rank39_refresh_busy	R	SoC core may initiate a rank39_refresh operation (refresh operation to rank 39) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank39_refresh is set to one. It goes low when the rank39_refresh operation is stored in the DDRCTL. It is recommended not to perform rank39_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank39_refresh operation ■ 1 - Indicates that rank39_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>39 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
6	rank38_refresh_busy	R	SoC core may initiate a rank38_refresh operation (refresh operation to rank 38) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank38_refresh is set to one. It goes low when the rank38_refresh operation is stored in the DDRCTL. It is recommended not to perform rank38_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank38_refresh operation ■ 1 - Indicates that rank38_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>38 **Programming Mode:** Dynamic
5	rank37_refresh_busy	R	SoC core may initiate a rank37_refresh operation (refresh operation to rank 37) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank37_refresh is set to one. It goes low when the rank37_refresh operation is stored in the DDRCTL. It is recommended not to perform rank37_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank37_refresh operation ■ 1 - Indicates that rank37_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>37 **Programming Mode:** Dynamic
4	rank36_refresh_busy	R	SoC core may initiate a rank36_refresh operation (refresh operation to rank 36) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank36_refresh is set to one. It goes low when the rank36_refresh operation is stored in the DDRCTL. It is recommended not to perform rank36_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank36_refresh operation ■ 1 - Indicates that rank36_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>36 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
3	rank35_refresh_busy	R	SoC core may initiate a rank35_refresh operation (refresh operation to rank 35) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank35_refresh is set to one. It goes low when the rank35_refresh operation is stored in the DDRCTL. It is recommended not to perform rank35_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank35_refresh operation ■ 1 - Indicates that rank35_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>35 **Programming Mode:** Dynamic
2	rank34_refresh_busy	R	SoC core may initiate a rank34_refresh operation (refresh operation to rank 34) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank34_refresh is set to one. It goes low when the rank34_refresh operation is stored in the DDRCTL. It is recommended not to perform rank34_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank34_refresh operation ■ 1 - Indicates that rank34_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>34 **Programming Mode:** Dynamic
1	rank33_refresh_busy	R	SoC core may initiate a rank33_refresh operation (refresh operation to rank 33) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank33_refresh is set to one. It goes low when the rank33_refresh operation is stored in the DDRCTL. It is recommended not to perform rank33_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank33_refresh operation ■ 1 - Indicates that rank33_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>33 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
0	rank32_refresh_busy	R	SoC core may initiate a rank32_refresh operation (refresh operation to rank 32) only if this signal is low. This signal goes high in the clock after OPREFCTRL1.rank32_refresh is set to one. It goes low when the rank32_refresh operation is stored in the DDRCTL. It is recommended not to perform rank32_refresh operations when this signal is high. - 0 - Indicates that the SoC core can initiate a rank32_refresh operation ■ 1 - Indicates that rank32_refresh operation has not been stored yet in the DDRCTL **Value After Reset:** 0x0 **Exists:** UMCTL2_NUM_LRANKS_TOTAL>32 **Programming Mode:** Dynamic

1.3.60 DBICTL

- **Description:** DM/DBI Control Register
- **Size:** 32 bits
- **Offset:** 0x11c94
- **Exists:** Always

This register is in block REGB_DDRC_CH1.

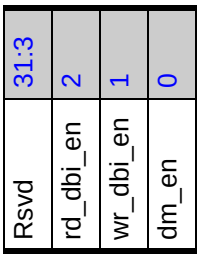


Table 1-241 Fields for Register: DBICTL

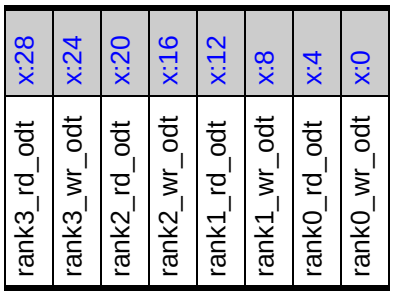
Bits	Name	Memory Access	Description
31:3			**Reserved Field:** Yes
2	rd_dbi_en	R/W	Read DBI enable signal in DDRC. ■ 0 - Read DBI is disabled. ■ 1 - Read DBI is enabled. This signal must be set the same value as DRAM's mode register. ■ DDR4: MR5 bit A12. When x4 devices are used, this signal must be set to 0. ■ DDR5: This signal must be set to 0. ■ LPDDR4/LPDDR5: MR3[6]. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1

Bits	Name	Memory Access	Description
1	wr_dbi_en	R/W	Write DBI enable signal in DDRC. ■ 0 - Write DBI is disabled. ■ 1 - Write DBI is enabled. This signal must be set the same value as DRAM's mode register. ■ DDR4: MR5 bit A11. When x4 devices are used, this signal must be set to 0. ■ DDR5: This signal must be set to 0. ■ LPDDR4/LPDDR5: MR3[7]. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 1
0	dm_en	R/W	DM enable signal in DDRC. ■ 0 - DM is disabled. ■ 1 - DM is enabled. This signal must be set the same logical value as DRAM's mode register. ■ DDR4: Set this to same value as MR5 bit A10. When x4 devices are used, this signal must be set to 0. ■ DDR5: Set this to same value as MR5[5]. When x4 devices are used, this signal must be set to 0. ■ LPDDR4/LPDDR5: Set this to inverted value of MR13[5] which is opposite polarity from this signal. **Value After Reset:** 0x1 **Exists:** Always **Volatile:** true **Programming Mode:** Static

1.3.61 ODTMAP

- **Description:** ODT/Rank Map Register
- **Size:** 32 bits
- **Offset:** 0x11c9c
- **Exists:** Always

This register is in block REGB_DDRC_CH1.



x:28	x:24	x:20	x:16	x:12	x:8	x:4	x:0
rank3_rd_odt	rank3_wr_odt	rank2_rd_odt	rank2_wr_odt	rank1_rd_odt	rank1_wr_odt	rank0_rd_odt	rank0_wr_odt

Table 1-242 Fields for Register: ODTMAP

Bits	Name	Memory Access	Description
x:28	rank3_rd_odt	R/W	Indicates which remote ODTs must be turned on during a read from rank 3. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. Present only in configurations that have 4 ranks **Value After Reset:** "(MEMC_NUM_RANKS>=4) ? 0x8 : 0x0" **Exists:** MEMC_NUM_RANKS==4 **Programming Mode:** Static **Range Variable[x]:** "MEMC_NUM_RANKS" + 27
x:24	rank3_wr_odt	R/W	Indicates which remote ODTs must be turned on during a write to rank 3. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. Present only in configurations that have 4 ranks **Value After Reset:** "(MEMC_NUM_RANKS>=4) ? 0x8 : 0x0" **Exists:** MEMC_NUM_RANKS==4 **Programming Mode:** Static **Range Variable[x]:** "MEMC_NUM_RANKS" + 23

Bits	Name	Memory Access	Description
x:20	rank2_rd_odt	R/W	Indicates which remote ODTs must be turned on during a read from rank 2. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. Present only in configurations that have 4 ranks **Value After Reset:** "(MEMC_NUM_RANKS>=4) ? 0x4 : 0x0" **Exists:** MEMC_NUM_RANKS==4 **Programming Mode:** Static **Range Variable[x]:** "MEMC_NUM_RANKS" + 19
x:16	rank2_wr_odt	R/W	Indicates which remote ODTs must be turned on during a write to rank 2. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. Present only in configurations that have 4 ranks **Value After Reset:** "(MEMC_NUM_RANKS>=4) ? 0x4 : 0x0" **Exists:** MEMC_NUM_RANKS==4 **Programming Mode:** Static **Range Variable[x]:** "MEMC_NUM_RANKS" + 15
x:12	rank1_rd_odt	R/W	Indicates which remote ODTs must be turned on during a read from rank 1. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. Present only in configurations that have 2 or more ranks **Value After Reset:** "(MEMC_NUM_RANKS>1) ? 0x2 : 0x0" **Exists:** MEMC_NUM_RANKS>1 **Programming Mode:** Static **Range Variable[x]:** "MEMC_NUM_RANKS" + 11

Bits	Name	Memory Access	Description
x:8	rank1_wr_odt	R/W	Indicates which remote ODTs must be turned on during a write to rank 1. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. Present only in configurations that have 2 or more ranks **Value After Reset:** "(MEMC_NUM_RANKS>1) ? 0x2 : 0x0" **Exists:** MEMC_NUM_RANKS>1 **Programming Mode:** Static **Range Variable[x]:** "MEMC_NUM_RANKS" + 7
x:4	rank0_rd_odt	R/W	Indicates which remote ODTs must be turned on during a read from rank 0. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Static **Range Variable[x]:** "MEMC_NUM_RANKS" + 3
x:0	rank0_wr_odt	R/W	Indicates which remote ODTs must be turned on during a write to rank 0. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Static **Range Variable[x]:** "MEMC_NUM_RANKS" - 1

1.3.62 INITTMG0

- **Description:** SDRAM Initialization Timing Register 0
- **Size:** 32 bits
- **Offset:** 0x11d00
- **Exists:** Always

This register is in block REGB_DDRC_CH1.

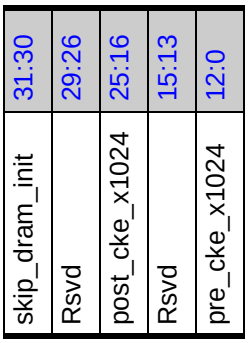


Table 1-243 Fields for Register: INITTMG0

Bits	Name	Memory Access	Description
31:30	skip_dram_init	R/W	If lower bit is enabled the SDRAM initialization routine is skipped. The upper bit decides what state the controller starts up in when reset is removed ■ 00 - SDRAM Initialization routine is run after power-up ■ 01 - SDRAM Initialization routine is skipped after power-up. Controller starts up in Normal Mode ■ 11 - SDRAM Initialization routine is skipped after power-up. Controller starts up in Self-refresh Mode ■ 10 - Reserved. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 2
29:26			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
25:16	post_cke_x1024	R/W	Cycles to wait after driving CKE (CS in DDR5) high to start the SDRAM initialization sequence. DDR5: tINIT4 of 2 us (min) - simulation only. LPDDR4: typically requires this to be programmed for a delay of 2 us. LPDDR5: Don't care Unit: Multiples of 1024 DRAM clock cycles. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x2 **Exists:** Always **Programming Mode:** Static
15:13			**Reserved Field:** Yes
12:0	pre_cke_x1024	R/W	Cycles to wait after reset before driving CKE (CS in DDR5) high to start the SDRAM initialization sequence. DDR5: tINIT3 of 4 ms (min) - simulation only. LPDDR4: tINIT3 of 2 ms (min) LPDDR5: Don't care (For LPDDR5, DDRCTL doesn't send the first PDX command to the SDRAM - Assumption is that the first PDX is issued as part of initialization performed by PHY) For DDR4 RDIMMs, this must include the time needed to satisfy tSTAB. Unit: Multiples of 1024 DRAM clock cycles. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x4e **Exists:** Always **Programming Mode:** Static

1.3.63 INITTMG1

- **Description:** SDRAM Initialization Timing Register 1
- **Size:** 32 bits
- **Offset:** 0x11d04
- **Exists:** Always

This register is in block REGB_DDRC_CH1.

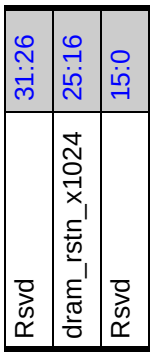


Table 1-244 Fields for Register: INITTMG1

Bits	Name	Memory Access	Description
31:26			**Reserved Field:** Yes
25:16	dram_rstn_x1024	R/W	Number of cycles to assert SDRAM reset signal during init sequence. For use with a Synopsys DDR PHY, this must be set to a minimum of 1. Unit: Multiples of 1024 DRAM clock cycles. Please refer to "Note 1" from "Notes on Timing Registers" at the start of "Register Descriptions" chapter for details on how to program this register field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
15:0			**Reserved Field:** Yes

1.4 REGB_ARB_PORTp Registers

1.4.1 PCCFG

- **Description:** Port Common Configuration Register.
- **Size:** 32 bits
- **Offset:** 0x20000+p*0x1000
- **Exists:** UMCTL2_INCL_ARB==1

This register is in block REGB_ARB_PORTp.
This register is only present in REGB_ARB_PORT0, therefore p=0 for the offset calculation

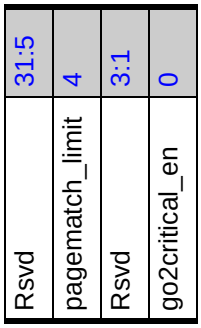


Table 1-245 Fields for Register: PCCFG

Bits	Name	Memory Access	Description
31:5			**Reserved Field:** Yes
4	pagematch_limit	R/W	Page match four limit. If set to 1, limits the number of consecutive same page DDRC transactions that can be granted by the Port Arbiter to four when Page Match feature is enabled. If set to 0, there is no limit imposed on number of consecutive same page DDRC transactions. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
3:1			**Reserved Field:** Yes
0	go2critical_en	R/W	If set to 1 (enabled), sets co_gs_go2critical_wr and co_gs_go2critical_lpr/co_gs_go2critical_hpr signals going to DDRC based on urgent input (awurgent, arurgent) coming from AXI master. If set to 0 (disabled), co_gs_go2critical_wr and co_gs_go2critical_lpr/co_gs_go2critical_hpr signals at DDRC are driven to 1b'0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.4.2 PCFGR

- **Description:** Configuration Read Register
- **Size:** 32 bits
- **Offset:** 0x20004+p*0x1000
- **Exists:** UMCTL2_INCL_ARB==1

This register is in block REGB_ARB_PORTp.

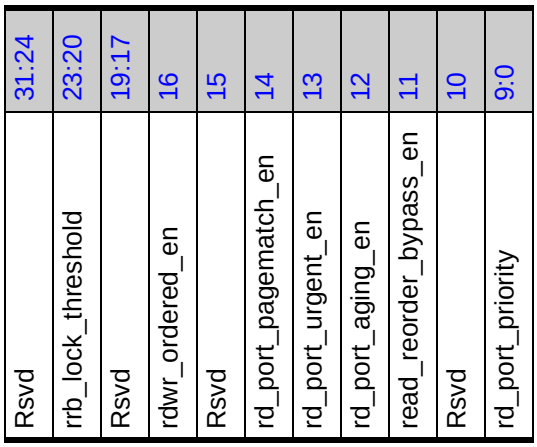


Table 1-246 Fields for Register: PCFGR

Bits	Name	Memory Access	Description
31:24			**Reserved Field:** Yes
23:20	rrb_lock_threshold	R/W	Specifies the RRB lock threshold in configurations that disable read data interleaving. Threshold is specified in terms of the HIF bursts that belong to the same AXI transaction. RRB locks onto VC only when this specified number of HIF bursts are returned by DDRC. RRB lock occurs earlier in cases where the axi transaction itself is shorter and the total number of corresponding HIF bursts are below the programmed threshold and all of them are returned by DDRC. When N is programmed in this field, the threshold will be set to N+1 bursts. Max thresholding is up to 16 bursts. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_RRB_THRESHOLD_EN_0==1 **Programming Mode:** Static
19:17			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
16	rdwr_ordered_en	R/W	Enable ordered read/writes. If set to 1, preserves the ordering between read transaction and write transaction issued to the same address, on a given port. In other words, the controller ensures that all same address read and write commands from the application port interface are transported to the DFI interface in the order of acceptance. This feature is useful in cases where software coherency is desired for masters issuing back-to-back read/write transactions without waiting for write/read responses. Note that this register has an effect only if necessary logic is instantiated via the UMCTL2_RDWR_ORDERED_n parameter. **Value After Reset:** 0x0 **Exists:** UMCTL2_A_RDWR_ORDERED_0==1 **Volatile:** true **Programming Mode:** Static
15			**Reserved Field:** Yes
14	rd_port_pagematch_en	R/W	If set to 1, enables the Page Match feature. If enabled, once a requesting port is granted, the port is continued to be granted if the following immediate commands are to the same memory page (same bank and same row). See also related PCCFG.pagematch_limit register. **Value After Reset:** "(MEMC_DDR4_EN==1) ? 0x0 : 0x1" **Exists:** Always **Programming Mode:** Static
13	rd_port_urgent_en	R/W	If set to 1, enables the AXI urgent sideband signal (arurgent). When enabled and arurgent is asserted by the master, that port becomes the highest priority and co_gs_go2critical_lpr/co_gs_go2critical_hpr signal to DDRC is asserted if enabled in PCCFG.go2critical_en register. Note that arurgent signal can be asserted anytime and as long as required which is independent of address handshaking (it is not associated with any particular command). **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
12	rd_port_aging_en	R/W	If set to 1, enables aging function for the read channel of the port. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Static

Bits	Name	Memory Access	Description
11	read_reorder_bypass_en	R/W	If set to 1, read transactions with ID not covered by any of the virtual channel ID mapping registers are not reordered. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_CH0_0==1 **Volatile:** true **Programming Mode:** Static
10			**Reserved Field:** Yes
9:0	rd_port_priority	R/W	Determines the initial load value of read aging counters. These counters will be parallel loaded after reset, or after each grant to the corresponding port. The aging counters down-count every clock cycle where the port is requesting but not granted. The higher significant 5-bits of the read aging counter sets the priority of the read channel of a given port. Port's priority will increase as the higher significant 5-bits of the counter starts to decrease. When the aging counter becomes 0, the corresponding port channel will have the highest priority level (timeout condition - Priority0). For multi-port configurations, the aging counters cannot be used to set port priorities when external dynamic priority inputs (arqos) are enabled (timeout is still applicable). For single port configurations, the aging counters are only used when they timeout (become 0) to force read-write direction switching. In this case, external dynamic priority input, arqos (for reads only) can still be used to set the DDRC read priority (2 priority levels: low priority read - LPR, high priority read - HPR) on a command by command basis. Note: The two LSBs of this register field are tied internally to 2'b00. **Value After Reset:** 0x1f **Exists:** Always **Programming Mode:** Static

1.4.3 PCFGW

- **Description:** Configuration Write Register
- **Size:** 32 bits
- **Offset:** 0x20008+p*0x1000
- **Exists:** UMCTL2_INCL_ARB==1

This register is in block REGB_ARB_PORTp.

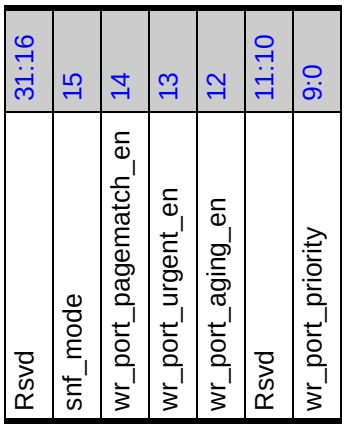


Table 1-247 Fields for Register: PCFGW

Bits	Name	Memory Access	Description
31:16			**Reserved Field:** Yes
15	snf_mode	R/W	If set to 1, enables Store & Forward Mode This bit controls the XPI port's Write Request Path. In 'Store & Forward' mode, XPI issues a HIF Write Request to the Port Arbiter only after all the respective HIF write Data beats is available. In non- 'Store & Forward' mode, XPI issues the HIF Write request to the Port Arbiter irrespective of the HIF Write Data availability within it. Values: 0 - Programmable Store & Forward is disabled 1 - Programmable Store & Forward is enabled If 'Read-Modify-Write' functionality is enabled through register programming - DBICTL.dm_en = 0/ ECCCFG0.reg_ecc_mode>0, 'Store & Forward' functionality can't be disabled. Hence, this register bit will be a don't care. **Value After Reset:** 0x1 **Exists:** DDRCTL_XPI_PROG_SNF==1 **Programming Mode:** Static

Bits	Name	Memory Access	Description
14	wr_port_pagematch_en	R/W	If set to 1, enables the Page Match feature. If enabled, once a requesting port is granted, the port is continued to be granted if the following immediate commands are to the same memory page (same bank and same row). See also related PCCFG.pagematch_limit register. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Static
13	wr_port_urgent_en	R/W	If set to 1, enables the AXI urgent sideband signal (awurgent). When enabled and awurgent is asserted by the master, that port becomes the highest priority and co_gs_go2critical_wr signal to DDRC is asserted if enabled in PCCFG.go2critical_en register. Note that awurgent signal can be asserted anytime and as long as required which is independent of address handshaking (it is not associated with any particular command). **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
12	wr_port_aging_en	R/W	If set to 1, enables aging function for the write channel of the port. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Static
11:10			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
9:0	wr_port_priority	R/W	Determines the initial load value of write aging counters. These counters will be parallel loaded after reset, or after each grant to the corresponding port. The aging counters down-count every clock cycle where the port is requesting but not granted. The higher significant 5-bits of the write aging counter sets the initial priority of the write channel of a given port. Port's priority will increase as the higher significant 5-bits of the counter starts to decrease. When the aging counter becomes 0, the corresponding port channel will have the highest priority level. For multi-port configurations, the aging counters cannot be used to set port priorities when external dynamic priority inputs (awqos) are enabled (timeout is still applicable). For single port configurations, the aging counters are only used when they timeout (become 0) to force read-write direction switching. Note: The two LSBs of this register field are tied internally to 2'b00. **Value After Reset:** 0x1f **Exists:** Always **Programming Mode:** Static

1.4.4 PCFGIDMASKCHc (for c = 0; c <= 15)

- **Description:** Channel c Configuration ID mask register
- **Size:** 32 bits
- **Offset:** 0x20010+p*0x1000+c*0x8
- **Exists:** UMCTL2_PORT_CH0_0==1 && UMCTL2_INCL_ARB==1

This register is in block REGB_ARB_PORTp.

Table 1-248 Fields for Register: PCFGIDMASKCHc (for c = 0; c <= 15)

Bits	Name	Memory Access	Description
x:0	id_mask	R/W	Determines the mask used in the ID mapping function for virtual channel m. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Static **Range Variable[x]:** "UMCTL2_A_ID_MAPW" - 1

1.4.5 PCFGIDVALUECHc (for c = 0; c <= 15)

- **Description:** Channel c Configuration ID value register
- **Size:** 32 bits
- **Offset:** 0x20014+p*0x1000+c*0x8
- **Exists:** UMCTL2_PORT_CH0_0==1 && UMCTL2_INCL_ARB==1

This register is in block REGB_ARB_PORTp.

Table 1-249 Fields for Register: PCFGIDVALUECHc (for c = 0; c <= 15)

Bits	Name	Memory Access	Description
x:0	id_value	R/W	Determines the value used in the ID mapping function for virtual channel m. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Static **Range Variable[x]:** "UMCTL2_A_ID_MAPW" - 1

1.4.6 PCTRL

- **Description:** Port Control Register
- **Size:** 32 bits
- **Offset:** 0x20090+p*0x1000
- **Exists:** Always

This register is in block REGB_ARB_PORTp.

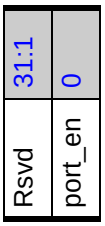


Table 1-250 Fields for Register: PCTRL

Bits	Name	Memory Access	Description
31:1			**Reserved Field:** Yes
0	port_en	R/W	Enables AXI port n. **Value After Reset:** "UMCTL2_PORT_EN_RESET_VALUE" **Exists:** Always **Programming Mode:** Dynamic

1.4.7 PCFGQOS0

- **Description:** Port n Read QoS Configuration Register 0
- **Size:** 32 bits
- **Offset:** 0x20094+p*0x1000
- **Exists:** Always

This register is in block REGB_ARB_PORTp.

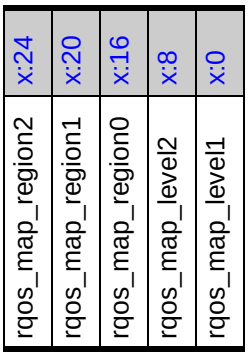


Table 1-251 Fields for Register: PCFGQOS0

Bits	Name	Memory Access	Description
x:24	rqos_map_region2	R/W	This bitfield indicates the traffic class of region2. For dual address queue configurations, region2 maps to the red address queue. Valid values are 1: VPR and 2: HPR only. When VPR support is disabled (UMCTL2_VPR_EN = 0) and traffic class of region2 is set to 1 (VPR), VPR traffic is aliased to LPR traffic. **Value After Reset:** 0x2 **Exists:** UMCTL2_A_USE2RAQ_0==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "UMCTL2_XPI_RQOS_RW" + 23

Bits	Name	Memory Access	Description
x:20	rqos_map_region1	R/W	This bitfield indicates the traffic class of region 1. Valid values are: ■ 0 : LPR ■ 1: VPR ■ 2: HPR For dual address queue configurations, region1 maps to the blue address queue. In this case, valid values are ■ 0: LPR ■ 1: VPR only When VPR support is disabled (UMCTL2_VPR_EN = 0) and traffic class of region 1 is set to 1 (VPR), VPR traffic is aliased to LPR traffic. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "UMCTL2_XPI_RQOS_RW" + 19
x:16	rqos_map_region0	R/W	This bitfield indicates the traffic class of region 0. Valid values are: ■ 0: LPR ■ 1: VPR ■ 2: HPR For dual address queue configurations, region 0 maps to the blue address queue. In this case, valid values are: 0: LPR and 1: VPR only. When VPR support is disabled (UMCTL2_VPR_EN = 0) and traffic class of region0 is set to 1 (VPR), VPR traffic is aliased to LPR traffic. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "UMCTL2_XPI_RQOS_RW" + 15

Bits	Name	Memory Access	Description
x:8	rqos_map_level2	R/W	Separation level2 indicating the end of region1 mapping; start of region1 is (level1 + 1). Possible values for level2 are (level1 + 1) to 14 which corresponds to arqos. Region2 starts from (level2 + 1) up to 15. Note that for PA, arqos values are used directly as port priorities, where the higher the value corresponds to higher port priority. All of the map_level* registers must be set to distinct values. **Value After Reset:** 0xe **Exists:** UMCTL2_A_USE2RAQ_0==1 **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "UMCTL2_XPI_RQOS_MLW" + 7
x:0	rqos_map_level1	R/W	Separation level1 indicating the end of region0 mapping; start of region0 is 0. Possible values for level1 are 0 to 13 (for dual RAQ) or 0 to 14 (for single RAQ) which corresponds to arqos. Note that for PA, arqos values are used directly as port priorities, where the higher the value corresponds to higher port priority. All of the map_level* registers must be set to distinct values. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "UMCTL2_XPI_RQOS_MLW" - 1

1.4.8 PCFGQOS1

- **Description:** Port n Read QoS Configuration Register 1
- **Size:** 32 bits
- **Offset:** 0x20098+p*0x1000
- **Exists:** Always

This register is in block REGB_ARB_PORTp.

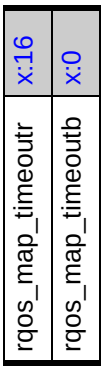


Table 1-252 Fields for Register: PCFGQOS1

Bits	Name	Memory Access	Description
x:16	rqos_map_timeoutr	R/W	Specifies the timeout value for transactions mapped to the red address queue. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "UMCTL2_XPI_RQOS_TW" + 15
x:0	rqos_map_timeoutb	R/W	Specifies the timeout value for transactions mapped to the blue address queue. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "UMCTL2_XPI_RQOS_TW" - 1

1.4.9 PCFGWQOS0

- **Description:** Port n Write QoS Configuration Register 0
- **Size:** 32 bits
- **Offset:** 0x2009c+p*0x1000
- **Exists:** Always

This register is in block REGB_ARB_PORTp.

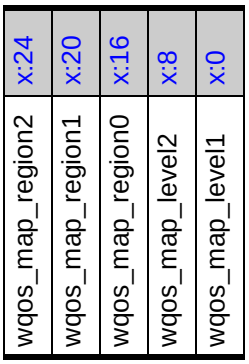


Table 1-253 Fields for Register: PCFGWQOS0

Bits	Name	Memory Access	Description
x:24	wqos_map_region2	R/W	This bitfield indicates the traffic class of region 2. Valid values are: 0: NPW, 1: VPW. When VPW support is disabled (UMCTL2_VPW_EN = 0) and traffic class of region 2 is set to 1 (VPW), VPW traffic is aliased to NPW traffic. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "UMCTL2_XPI_WQOS_RW" + 23
x:20	wqos_map_region1	R/W	This bitfield indicates the traffic class of region 1. Valid values are: 0: NPW, 1: VPW. When VPW support is disabled (UMCTL2_VPW_EN = 0) and traffic class of region 1 is set to 1 (VPW), VPW traffic is aliased to NPW traffic. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "UMCTL2_XPI_WQOS_RW" + 19

Bits	Name	Memory Access	Description
x:16	wqos_map_region0	R/W	This bitfield indicates the traffic class of region 0. Valid values are: 0: NPW, 1: VPW. When VPW support is disabled (UMCTL2_VPW_EN = 0) and traffic class of region 0 is set to 1 (VPW), VPW traffic is aliased to NPW traffic. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "UMCTL2_XPI_WQOS_RW" + 15
x:8	wqos_map_level2	R/W	Separation level2 indicating the end of region1 mapping; start of region1 is (level1 + 1). Possible values for level2 are (level1 + 1) to 14 which corresponds to awqos. Region2 starts from (level2 + 1) up to 15. Note that for PA, awqos values are used directly as port priorities, where the higher the value corresponds to higher port priority. All of the map_level* registers must be set to distinct values. **Value After Reset:** 0xe **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "UMCTL2_XPI_WQOS_MLW" + 7
x:0	wqos_map_level1	R/W	Separation level indicating the end of region0 mapping; start of region0 is 0. Possible values for level1 are 0 to 13 which corresponds to awqos. Note that for PA, awqos values are used directly as port priorities, where the higher the value corresponds to higher port priority. All of the map_level* registers must be set to distinct values. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "UMCTL2_XPI_WQOS_MLW" - 1

1.4.10 PCFGWQOS1

- **Description:** Port n Write QoS Configuration Register 1
- **Size:** 32 bits
- **Offset:** 0x200a0+p*0x1000
- **Exists:** Always

This register is in block REGB_ARB_PORTp.

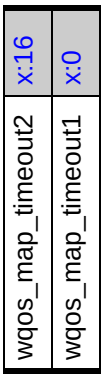


Table 1-254 Fields for Register: PCFGWQOS1

Bits	Name	Memory Access	Description
x:16	wqos_map_timeout2	R/W	Specifies the timeout value for write transactions in region 2. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "UMCTL2_XPI_WQOS_TW" + 15
x:0	wqos_map_timeout1	R/W	Specifies the timeout value for write transactions in region 0 and 1. **Value After Reset:** 0x0 **Exists:** Always **Volatile:** true **Programming Mode:** Quasi-dynamic Group 3 **Range Variable[x]:** "UMCTL2_XPI_WQOS_TW" - 1

1.4.11 SARBASEs (for s = 0; s <= 3)

- **Description:** SAR Base Address Register s.
- **Size:** 32 bits
- **Offset:** 0x200c0+p*0x1000+s*0x8
- **Exists:** UMCTL2_A_SAR_0==1 && UMCTL2_INCL_ARB==1

This register is in block REGB_ARB_PORTp.
This register is only present in REGB_ARB_PORT0, therefore p=0 for the offset calculation

Table 1-255 Fields for Register: SARBASEs (for s = 0; s <= 3)

Bits	Name	Memory Access	Description
x:0	base_addr	R/W	Base address for address region n specified as awaddr[UMCTL2_A_ADDRW-1:x] and araddr[UMCTL2_A_ADDRW-1:x] where x is determined by the minimum block size parameter UMCTL2_SARMINSIZE: (x=log2(block size)). **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static **Range Variable[x]:** "UMCTL2_AXI_SAR_REG_BW" - 1

1.4.12 SARSIZEs (for s = 0; s <= 3)

- **Description:** SAR Size Register s.
- **Size:** 32 bits
- **Offset:** 0x200c4+p*0x1000+s*0x8
- **Exists:** UMCTL2_A_SAR_0==1 && UMCTL2_INCL_ARB==1

This register is in block REGB_ARB_PORTp.
This register is only present in REGB_ARB_PORT0, therefore p=0 for the offset calculation

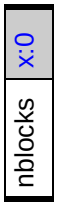


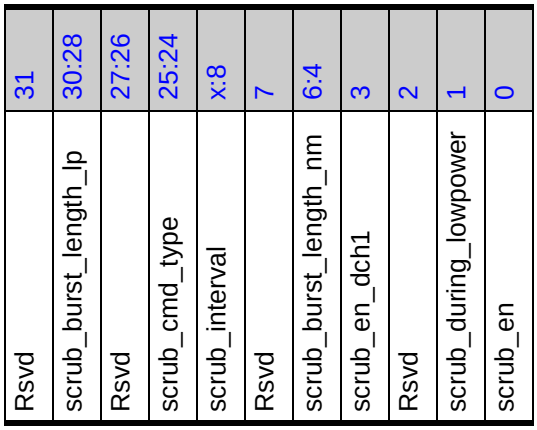
Table 1-256 Fields for Register: SARSIZEs (for s = 0; s <= 3)

Bits	Name	Memory Access	Description
x:0	nblocks	R/W	Number of blocks for address region n. This register determines the total size of the region in multiples of minimum block size as specified by the hardware parameter UMCTL2_SARMINSIZE. The register value is encoded as number of blocks = nblocks + 1. For example, if register is programmed to 0, region will have 1 block. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static **Range Variable[x]:** "UMCTL2_AXI_SAR_SW" - 1

1.4.13 SBRCTL

- **Description:** Scrubber Control Register
- **Size:** 32 bits
- **Offset:** 0x200e0+p*0x1000
- **Exists:** UMCTL2_SBR_EN_1==1

This register is in block REGB_ARB_PORTp.
This register is only present in REGB_ARB_PORT0, therefore p=0 for the offset calculation



31	30:28	27:26	25:24	x:8	7	6:4	3	2	1	0
Rsvd	scrub_burst_length_lp	Rsvd	scrub_cmd_type	scrub_interval	Rsvd	scrub_burst_length_nm	scrub_en_dch1	Rsvd	scrub_during_lowpower	scrub_en

Table 1-257 Fields for Register: SBRCTL

Bits	Name	Memory Access	Description
31			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
30:28	scrub_burst_length_lp	R/W	Scrub burst length in Low Power mode - Determines the number of back-to-back scrub read commands that can be issued together when the controller is in one of the HW controlled low power modes with Sideband ECC and Inline ECC. - During these modes, the period of the scrub burst becomes "scrub_burst_length_lp*scrub_interval" cycles. Valid values are (Sideband ECC): ■ 1: 1 read, ■ 2: 4 reads, ■ 3: 16 reads, ■ 4: 64 reads, ■ 5: 256 reads, ■ 6: 1024 reads. (Inline ECC): ■ 1: 8 reads, ■ 2: 16 reads, ■ 3: 32 reads. To program a new value to this register field, first disable Scrubber by setting SBRCTL.scrub_en = 0. Program the new value.Enable Scrubber by setting SBRCTL.scrub_en = 1. **Value After Reset:** 0x1 **Exists:** UMCTL2_SBR_EN_1==1 **Programming Mode:** Dynamic
27:26			**Reserved Field:** Yes
25:24	scrub_cmd_type	R/W	This field determines the kind of traffic scrubber must generate. ■ 00: Read - Only periodic reads will be generated ■ 01: Write - Only back to back initialization writes will be generated. SBRCTL.scrub_interval must be programmed to 0. ■ 10: Read Modify Write - only periodic RMWs will be generated. ■ 11: reserved. **Value After Reset:** 0x0 **Exists:** UMCTL2_SBR_EN_1==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
x:8	scrub_interval	R/W	Scrub interval. (N x scrub_interval) number of clock cycles between two scrub read commands, where N is the granularity. If set to 0, scrub commands are issued back-to-back. This mode of operation (scrub_interval=0) can typically be used for scrubbing the full range of memory at once before or after SW controlled low power operations. After completing the full range of scrub while scrub_interval=0, scrub_done register is set and sbr_done_intr interrupt signal is asserted. This mode can't be used with Inline ECC: If MEMC_INLINE_ECC is 1 and scrub_interval is programmed to 0, then RMW logic inside scrubber is disabled. New programmed value will take effect only after scrubber is disabled by programming scrub_en to 0. Unit: Multiples of 256 sbr_clk cycles in Sideband ECC configurations and 512 sbr_clk cycles in Inline ECC Configurations. **Value After Reset:** 0xff **Exists:** UMCTL2_SBR_EN_1==1 **Programming Mode:** Dynamic **Range Variable[x]:** "UMCTL2_REG_SCRUB_INTERVALW" + 7
7			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
6:4	scrub_burst_length_nm	R/W	Scrub burst length in normal mode. - Determines the number of back-to-back scrub read commands that can be issued together when the controller is in normal operation in Inline ECC & Sideband ECC. - The period of the scrub burst becomes "scrub_burst_length_nm*scrub_interval" cycles. During normal operation mode of the controller with Sideband ECC (not in power-down or self refresh), scrub_burst_length_nm is ignored and only one scrub command is generated. Valid values are (Sideband ECC): ■ 1: 1 read (Inline ECC): ■ 1: 8 reads, ■ 2: 16 reads, ■ 3: 32 reads. In Sideband ECC, software must ensure that the scrub_burst_length_nm is programmed to the value of 1. Other values are not supported. To program a new value to this register field, first disable Scrubber by setting SBRCTL.scrub_en = 0. Program the new value.Enable Scrubber by setting SBRCTL.scrub_en = 1. **Value After Reset:** 0x1 **Exists:** UMCTL2_SBR_EN_1==1 **Programming Mode:** Dynamic
3	scrub_en_dch1	R/W	Enable ECC scrubber for channel 1. If set to 1, enables the scrubber to generate background read commands after the memories are initialized. If set to 0, disables the scrubber, resets the address generator to 0 and clears the scrubber status. This bitfield must be accessed separately from the other bitfields in this register. **Value After Reset:** 0x0 **Exists:** UMCTL2_SBR_EN_1==1 && UMCTL2_DUAL_CHANNEL==1 **Programming Mode:** Dynamic
2			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
1	scrub_during_lowpower	R/W	Continue scrubbing during low power. If set to 1, burst of scrubs will be issued in HW controlled low power modes. There are two such modes: automatically initiated by idleness or initiated by Hardware low power interface. If set to 0, the scrubber will not attempt to send commands while the DDRC is in HW controlled low power modes. In this case, the scrubber will remember the last address issued and will automatically continue from there when the DDRC exits the LP mode. **Value After Reset:** 0x0 **Exists:** UMCTL2_SBR_EN_1==1 **Programming Mode:** Dynamic
0	scrub_en	R/W	Enable ECC scrubber. If set to 1, enables the scrubber to generate background read commands after the memories are initialized. If set to 0, disables the scrubber, resets the address generator to 0 and clears the scrubber status. This bitfield must be accessed separately from the other bitfields in this register. **Value After Reset:** 0x0 **Exists:** UMCTL2_SBR_EN_1==1 **Programming Mode:** Dynamic

1.4.14 SBRSTAT

- **Description:** Scrubber Status Register
- **Size:** 32 bits
- **Offset:** 0x200e4+p*0x1000
- **Exists:** UMCTL2_SBR_EN_1==1

This register is in block REGB_ARB_PORTp.
This register is only present in REGB_ARB_PORT0, therefore p=0 for the offset calculation

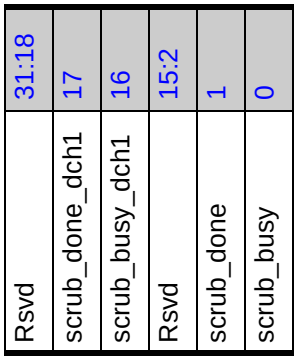


Table 1-258 Fields for Register: SBRSTAT

Bits	Name	Memory Access	Description
31:18			**Reserved Field:** Yes
17	scrub_done_dch1	R	Scrubber done for channel 1. Controller sets this bit to 1, after full range of addresses are scrubbed once while scrub_interval is set to 0. Cleared if scrub_en is set to 0 (scrubber disabled) or scrub_interval is set to a non-zero value for normal scrub operation. The interrupt signal, sbr_done_intr, is equivalent to this status bitfield. **Value After Reset:** 0x0 **Exists:** UMCTL2_SBR_EN_1==1 && UMCTL2_DUAL_CHANNEL==1 **Programming Mode:** Dynamic
16	scrub_busy_dch1	R	Scrubber busy for channel 1. Controller sets this bit to 1 when the scrubber logic has outstanding read commands being executed. Cleared when there are no active outstanding scrub reads in the system. **Value After Reset:** 0x0 **Exists:** UMCTL2_SBR_EN_1==1 && UMCTL2_DUAL_CHANNEL==1 **Programming Mode:** Dynamic
15:2			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
1	scrub_done	R	Scrubber done. Controller sets this bit to 1, after full range of addresses are scrubbed once while scrub_interval is set to 0. Cleared if scrub_en is set to 0 (scrubber disabled) or scrub_interval is set to a non-zero value for normal scrub operation. The interrupt signal, sbr_done_intr, is equivalent to this status bitfield. **Value After Reset:** 0x0 **Exists:** UMCTL2_SBR_EN_1==1 **Programming Mode:** Dynamic
0	scrub_busy	R	Scrubber busy. Controller sets this bit to 1 when the scrubber logic has outstanding read commands being executed. Cleared when there are no active outstanding scrub reads in the system. **Value After Reset:** 0x0 **Exists:** UMCTL2_SBR_EN_1==1 **Programming Mode:** Dynamic

1.4.15 SBRWDATA0

- **Description:** Scrubber Write Data Pattern0
- **Size:** 32 bits
- **Offset:** 0x200e8+p*0x1000
- **Exists:** UMCTL2_SBR_EN_1==1

This register is in block REGB_ARB_PORTp.
This register is only present in REGB_ARB_PORT0, therefore p=0 for the offset calculation

Table 1-259 Fields for Register: SBRWDATA0

Bits	Name	Memory Access	Description
31:0	scrub_pattern0	R/W	ECC Scrubber write data pattern for data bus[31:0] **Value After Reset:** 0x0 **Exists:** UMCTL2_SBR_EN_1==1 **Programming Mode:** Dynamic

1.4.16 SBRWDATA1

- **Description:** Scrubber Write Data Pattern1
- **Size:** 32 bits
- **Offset:** 0x200ec+p*0x1000
- **Exists:** UMCTL2_SBR_EN_1==1 && MEMC_DRAM_DATA_WIDTH==64

This register is in block REGB_ARB_PORTp.
This register is only present in REGB_ARB_PORT0, therefore p=0 for the offset calculation

Table 1-260 Fields for Register: SBRWDATA1

Bits	Name	Memory Access	Description
31:0	scrub_pattern1	R/W	ECC Scrubber write data pattern for data bus[63:32] **Value After Reset:** 0x0 **Exists:** UMCTL2_SBR_EN_1==1 && MEMC_DRAM_DATA_WIDTH==64 **Programming Mode:** Dynamic

1.4.17 SBRSTART0

- **Description:** Scrubber Start Address Mask Register 0
- **Size:** 32 bits
- **Offset:** 0x200f0+p*0x1000
- **Exists:** UMCTL2_SBR_EN_1==1

This register is in block REGB_ARB_PORTp.
This register is only present in REGB_ARB_PORT0, therefore p=0 for the offset calculation

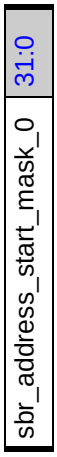


Table 1-261 Fields for Register: SBRSTART0

Bits	Name	Memory Access	Description
31:0	sbr_address_start_mask_0	R/W	sbr_address_start_mask_0 holds the bits [31:0] of the starting address the ECC scrubber generates. The register must be programmed as explained in Address Configuration in "Scrubber" section of DesignWare Cores DDR5/4 Memory Controller Databook. The scrubber address registers are changed only when the scrubber is disabled (SBRCTL.scrub_en = 0) and there are no scrubber commands in progress (SBRSTAT.scrub_busy = 0). **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.4.18 SBRSTART1

- **Description:** Scrubber Start Address Mask Register 1
- **Size:** 32 bits
- **Offset:** 0x200f4+p*0x1000
- **Exists:** UMCTL2_SBR_EN_1==1

This register is in block REGB_ARB_PORTp.
This register is only present in REGB_ARB_PORT0, therefore p=0 for the offset calculation

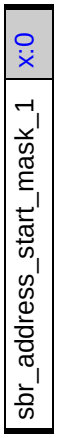


Table 1-262 Fields for Register: SBRSTART1

Bits	Name	Memory Access	Description
x:0	sbr_address_start_mask_1	R/W	sbr_address_start_mask_1 holds bits [MEMC_HIF_ADDR_WIDTH_MAX-1:32] of the starting address the ECC scrubber generates.The register must be programmed as explained in Address Configuration in "Scrubber" section of DesignWare Cores DDR5/4 Memory Controller Databook. The scrubber address registers are changed only when the scrubber is disabled (SBRCTL.scrub_en = 0) and there are no scrubber commands in progress (SBRSTAT.scrub_busy = 0). **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_HIF_ADDR_WIDTH_MAX - 32" - 1

1.4.19 SBRRANGE0

- **Description:** Scrubber Address Range Mask Register 0
- **Size:** 32 bits
- **Offset:** 0x200f8+p*0x1000
- **Exists:** UMCTL2_SBR_EN_1==1

This register is in block REGB_ARB_PORTp.
This register is only present in REGB_ARB_PORT0, therefore p=0 for the offset calculation

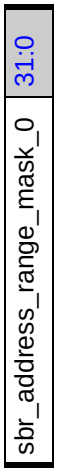


Table 1-263 Fields for Register: SBRRANGE0

Bits	Name	Memory Access	Description
31:0	sbr_address_range_mask_0	R/W	sbr_address_range_mask_0 holds the bits [31:0] of the scrubber address range mask. The scrubber address range mask limits the address range that the ECC scrubber can generate.The register must be programmed as explained in Address Configuration in "Scrubber" section of DesignWare Cores DDR5/4 Memory Controller Databook. The scrubber address registers are changed only when the scrubber is disabled (SBRCTL.scrub_en = 0) and there are no scrubber commands in progress (SBRSTAT.scrub_busy = 0). **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.4.20 SBRRANGE1

- **Description:** Scrubber Address Range Mask Register 1
- **Size:** 32 bits
- **Offset:** 0x200fc+p*0x1000
- **Exists:** UMCTL2_SBR_EN_1==1

This register is in block REGB_ARB_PORTp.
This register is only present in REGB_ARB_PORT0, therefore p=0 for the offset calculation

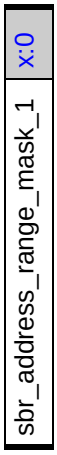


Table 1-264 Fields for Register: SBRRANGE1

Bits	Name	Memory Access	Description
x:0	sbr_address_range_mask_1	R/W	sbr_address_range_mask_1 holds the bits [MEMC_HIF_ADDR_WIDTH_MAX-1:32] of the scrubber address range mask. The scrubber address range mask limits the address range that the ECC scrubber can generate.The register must be programmed as explained in Address Configuration in "Scrubber" section of DesignWare Cores DDR5/4 Memory Controller Databook. The scrubber address registers are changed only when the scrubber is disabled (SBRCTL.scrub_en = 0) and there are no scrubber commands in progress (SBRSTAT.scrub_busy = 0). **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_HIF_ADDR_WIDTH_MAX - 32" - 1

1.4.21 SBRSTART0DCH1

- **Description:** Scrubber Start Address Mask Register 0 for Data Channel 1
- **Size:** 32 bits
- **Offset:** 0x20100+p*0x1000
- **Exists:** UMCTL2_SBR_EN_1==1 && UMCTL2_DUAL_CHANNEL==1

This register is in block REGB_ARB_PORTp.
This register is only present in REGB_ARB_PORT0, therefore p=0 for the offset calculation

Table 1-265 Fields for Register: SBRSTART0DCH1

Bits	Name	Memory Access	Description
31:0	sbr_address_start_mask_dch1_0	R/W	sbr_address_start_mask_dch1_0 holds the bits [31:0] of the starting address the ECC scrubber generates for data channel 1.The register must be programmed as explained in Address Configuration in "Scrubber" section of DesignWare Cores DDR5/4 Memory Controller Databook. The scrubber address registers are changed only when the scrubber is disabled (SBRCTL.scrub_en = 0) and there are no scrubber commands in progress (SBRSTAT.scrub_busy = 0). **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.4.22 SBRSTART1DCH1

- **Description:** Scrubber Start Address Mask Register 1 for Data Channel 1
- **Size:** 32 bits
- **Offset:** 0x20104+p*0x1000
- **Exists:** UMCTL2_SBR_EN_1==1 && UMCTL2_DUAL_CHANNEL==1

This register is in block REGB_ARB_PORTp.
This register is only present in REGB_ARB_PORT0, therefore p=0 for the offset calculation

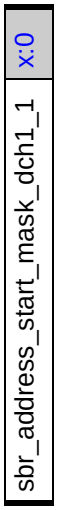


Table 1-266 Fields for Register: SBRSTART1DCH1

Bits	Name	Memory Access	Description
x:0	sbr_address_start_mask_dch1_1	R/W	sbr_address_start_mask_dch1_1 holds bits [MEMC_HIF_ADDR_WIDTH_MAX-1:32] of the starting address the ECC scrubber generates for data channel 1.The register must be programmed as explained in Address Configuration in "Scrubber" section of DesignWare Cores DDR5/4 Memory Controller Databook. The scrubber address registers are changed only when the scrubber is disabled (SBRCTL.scrub_en = 0) and there are no scrubber commands in progress (SBRSTAT.scrub_busy = 0). **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_HIF_ADDR_WIDTH_MAX - 32" - 1

1.4.23 SBRRANGE0DCH1

- **Description:** Scrubber Address Range Mask Register 0 for Data Channel 1
- **Size:** 32 bits
- **Offset:** 0x20108+p*0x1000
- **Exists:** UMCTL2_SBR_EN_1==1 && UMCTL2_DUAL_CHANNEL==1

This register is in block REGB_ARB_PORTp.
This register is only present in REGB_ARB_PORT0, therefore p=0 for the offset calculation

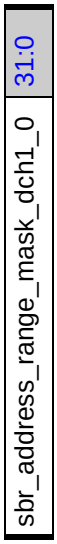


Table 1-267 Fields for Register: SBRRANGE0DCH1

Bits	Name	Memory Access	Description
31:0	sbr_address_range_mask_dch1_0	R/W	sbr_address_range_mask_dch1_0 holds bits [31:0] of the scrubber address range mask for data channel 1. The scrubber address range mask limits the address range that the ECC scrubber can generate.The register must be programmed as explained in Address Configuration in "Scrubber" section of DesignWare Cores DDR5/4 Memory Controller Databook. The scrubber address registers are changed only when the scrubber is disabled (SBRCTL.scrub_en = 0) and there are no scrubber commands in progress (SBRSTAT.scrub_busy = 0). **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.4.24 SBRRANGE1DCH1

- **Description:** Scrubber Address Range Mask Register 1 for Data Channel 1
- **Size:** 32 bits
- **Offset:** 0x2010c+p*0x1000
- **Exists:** UMCTL2_SBR_EN_1==1 && UMCTL2_DUAL_CHANNEL==1

This register is in block REGB_ARB_PORTp.
This register is only present in REGB_ARB_PORT0, therefore p=0 for the offset calculation

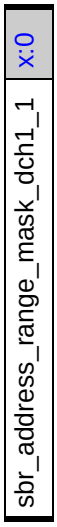


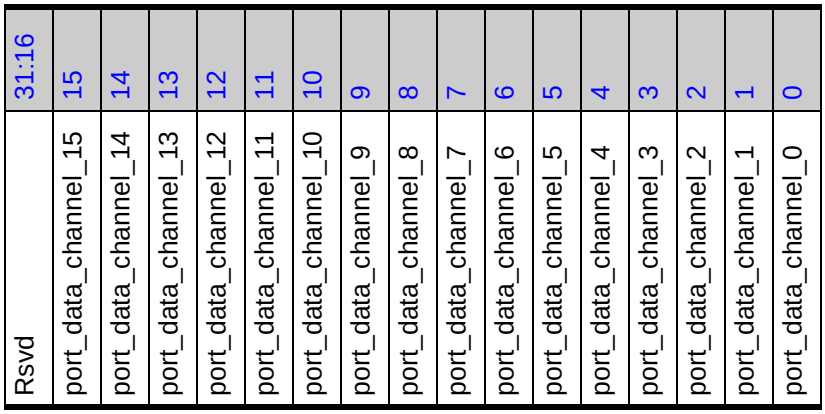
Table 1-268 Fields for Register: SBRRANGE1DCH1

Bits	Name	Memory Access	Description
x:0	sbr_address_range_mask_dch1_1	R/W	sbr_address_range_mask_dch1_1 holds bits [MEMC_HIF_ADDR_WIDTH_MAX-1:32] of the scrubber address range mask for data channel 1. The scrubber address range mask limits the address range that the ECC scrubber can generate. The register must be programmed as explained in Address Configuration in "Scrubber" section of DesignWare Cores DDR5/4 Memory Controller Databook. The scrubber address registers are changed only when the scrubber is disabled (SBRCTL.scrub_en = 0) and there are no scrubber commands in progress (SBRSTAT.scrub_busy = 0). **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "MEMC_HIF_ADDR_WIDTH_MAX - 32" - 1

1.4.25 PDCH

- **Description:** Port Data Channel
- **Size:** 32 bits
- **Offset:** 0x20110+p*0x1000
- **Exists:** UMCTL2_DUAL_CHANNEL==1 && UMCTL2_DATA_CHANNEL_INTERLEAVE_EN_1==0

This register is in block REGB_ARB_PORTp.
This register is only present in REGB_ARB_PORT0, therefore p=0 for the offset calculation



31:16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Rsvd	port_data_channel_15	port_data_channel_14	port_data_channel_13	port_data_channel_12	port_data_channel_11	port_data_channel_10	port_data_channel_9	port_data_channel_8	port_data_channel_7	port_data_channel_6	port_data_channel_5	port_data_channel_4	port_data_channel_3	port_data_channel_2	port_data_channel_1	port_data_channel_0

Table 1-269 Fields for Register: PDCH

Bits	Name	Memory Access	Description
31:16			**Reserved Field:** Yes
15	port_data_channel_15	R/W	Static data channel assignment for port 15: ■ 0 selects Channel 0 ■ 1 selects Channel 1 **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_15==1 && UMCTL2_DUAL_CHANNEL==1 && UMCTL2_DATA_CHANNEL_INTERLEAVE_EN_1==0 **Programming Mode:** Static
14	port_data_channel_14	R/W	Static data channel assignment for port 14: ■ 0 selects Channel 0 ■ 1 selects Channel 1 **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_14==1 && UMCTL2_DUAL_CHANNEL==1 && UMCTL2_DATA_CHANNEL_INTERLEAVE_EN_1==0 **Programming Mode:** Static

Bits	Name	Memory Access	Description
13	port_data_channel_13	R/W	Static data channel assignment for port 13: ■ 0 selects Channel 0 ■ 1 selects Channel 1 **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_13==1 && UMCTL2_DUAL_CHANNEL==1 && UMCTL2_DATA_CHANNEL_INTERLEAVE_EN_1==0 **Programming Mode:** Static
12	port_data_channel_12	R/W	Static data channel assignment for port 12: ■ 0 selects Channel 0 ■ 1 selects Channel 1 **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_12==1 && UMCTL2_DUAL_CHANNEL==1 && UMCTL2_DATA_CHANNEL_INTERLEAVE_EN_1==0 **Programming Mode:** Static
11	port_data_channel_11	R/W	Static data channel assignment for port 11: ■ 0 selects Channel 0 ■ 1 selects Channel 1 **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_11==1 && UMCTL2_DUAL_CHANNEL==1 && UMCTL2_DATA_CHANNEL_INTERLEAVE_EN_1==0 **Programming Mode:** Static
10	port_data_channel_10	R/W	Static data channel assignment for port 10: ■ 0 selects Channel 0 ■ 1 selects Channel 1 **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_10==1 && UMCTL2_DUAL_CHANNEL==1 && UMCTL2_DATA_CHANNEL_INTERLEAVE_EN_1==0 **Programming Mode:** Static
9	port_data_channel_9	R/W	Static data channel assignment for port 9: ■ 0 selects Channel 0 ■ 1 selects Channel 1 **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_9==1 && UMCTL2_DUAL_CHANNEL==1 && UMCTL2_DATA_CHANNEL_INTERLEAVE_EN_1==0 **Programming Mode:** Static

Bits	Name	Memory Access	Description
8	port_data_channel_8	R/W	Static data channel assignment for port 8: ■ 0 selects Channel 0 ■ 1 selects Channel 1 **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_8==1 && UMCTL2_DUAL_CHANNEL==1 && UMCTL2_DATA_CHANNEL_INTERLEAVE_EN_1==0 **Programming Mode:** Static
7	port_data_channel_7	R/W	Static data channel assignment for port 7: ■ 0 selects Channel 0 ■ 1 selects Channel 1 **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_7==1 && UMCTL2_DUAL_CHANNEL==1 && UMCTL2_DATA_CHANNEL_INTERLEAVE_EN_1==0 **Programming Mode:** Static
6	port_data_channel_6	R/W	Static data channel assignment for port 6: ■ 0 selects Channel 0 ■ 1 selects Channel 1 **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_6==1 && UMCTL2_DUAL_CHANNEL==1 && UMCTL2_DATA_CHANNEL_INTERLEAVE_EN_1==0 **Programming Mode:** Static
5	port_data_channel_5	R/W	Static data channel assignment for port 5: ■ 0 selects Channel 0 ■ 1 selects Channel 1 **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_5==1 && UMCTL2_DUAL_CHANNEL==1 && UMCTL2_DATA_CHANNEL_INTERLEAVE_EN_1==0 **Programming Mode:** Static
4	port_data_channel_4	R/W	Static data channel assignment for port 4: ■ 0 selects Channel 0 ■ 1 selects Channel 1 **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_4==1 && UMCTL2_DUAL_CHANNEL==1 && UMCTL2_DATA_CHANNEL_INTERLEAVE_EN_1==0 **Programming Mode:** Static

Bits	Name	Memory Access	Description
3	port_data_channel_3	R/W	Static data channel assignment for port 3: ■ 0 selects Channel 0 ■ 1 selects Channel 1 **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_3==1 && UMCTL2_DUAL_CHANNEL==1 && UMCTL2_DATA_CHANNEL_INTERLEAVE_EN_1==0 **Programming Mode:** Static
2	port_data_channel_2	R/W	Static data channel assignment for port 2: ■ 0 selects Channel 0 ■ 1 selects Channel 1 **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_2==1 && UMCTL2_DUAL_CHANNEL==1 && UMCTL2_DATA_CHANNEL_INTERLEAVE_EN_1==0 **Programming Mode:** Static
1	port_data_channel_1	R/W	Static data channel assignment for port 1: ■ 0 selects Channel 0 ■ 1 selects Channel 1 **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_1==1 && UMCTL2_DUAL_CHANNEL==1 && UMCTL2_DATA_CHANNEL_INTERLEAVE_EN_1==0 **Programming Mode:** Static
0	port_data_channel_0	R/W	Static data channel assignment for port 0: ■ 0 selects Channel 0 ■ 1 selects Channel 1 **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_0==1 && UMCTL2_DUAL_CHANNEL==1 && UMCTL2_DATA_CHANNEL_INTERLEAVE_EN_1==0 **Programming Mode:** Static

1.4.26 PSTAT

- **Description:** Port Status Register
- **Size:** 32 bits
- **Offset:** 0x20114+p*0x1000
- **Exists:** Always

This register is in block REGB_ARB_PORTp.
This register is only present in REGB_ARB_PORT0, therefore p=0 for the offset calculation

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
wr_port_busy_15	wr_port_busy_14	wr_port_busy_13	wr_port_busy_12	wr_port_busy_11	wr_port_busy_10	wr_port_busy_9	wr_port_busy_8	wr_port_busy_7	wr_port_busy_6	wr_port_busy_5	wr_port_busy_4	wr_port_busy_3	wr_port_busy_2	wr_port_busy_1	wr_port_busy_0	rd_port_busy_15	rd_port_busy_14	rd_port_busy_13	rd_port_busy_12	rd_port_busy_11	rd_port_busy_10	rd_port_busy_9	rd_port_busy_8	rd_port_busy_7	rd_port_busy_6	rd_port_busy_5	rd_port_busy_4	rd_port_busy_3	rd_port_busy_2	rd_port_busy_1	rd_port_busy_0

Table 1-270 Fields for Register: PSTAT

Bits	Name	Memory Access	Description
31	wr_port_busy_15	R	Indicates if there are outstanding writes for AXI/CHI port 15. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_15==1 **Programming Mode:** Dynamic
30	wr_port_busy_14	R	Indicates if there are outstanding writes for AXI/CHI port 14. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_14==1 **Programming Mode:** Dynamic
29	wr_port_busy_13	R	Indicates if there are outstanding writes for AXI/CHI port 13. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_13==1 **Programming Mode:** Dynamic
28	wr_port_busy_12	R	Indicates if there are outstanding writes for AXI/CHI port 12. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_12==1 **Programming Mode:** Dynamic
27	wr_port_busy_11	R	Indicates if there are outstanding writes for AXI/CHI port 11. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_11==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
26	wr_port_busy_10	R	Indicates if there are outstanding writes for AXI/CHI port 10. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_10==1 **Programming Mode:** Dynamic
25	wr_port_busy_9	R	Indicates if there are outstanding writes for AXI/CHI port 9. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_9==1 **Programming Mode:** Dynamic
24	wr_port_busy_8	R	Indicates if there are outstanding writes for AXI/CHI port 8. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_8==1 **Programming Mode:** Dynamic
23	wr_port_busy_7	R	Indicates if there are outstanding writes for AXI/CHI port 7. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_7==1 **Programming Mode:** Dynamic
22	wr_port_busy_6	R	Indicates if there are outstanding writes for AXI/CHI port 6. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_6==1 **Programming Mode:** Dynamic
21	wr_port_busy_5	R	Indicates if there are outstanding writes for AXI/CHI port 5. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_5==1 **Programming Mode:** Dynamic
20	wr_port_busy_4	R	Indicates if there are outstanding writes for AXI/CHI port 4. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_4==1 **Programming Mode:** Dynamic
19	wr_port_busy_3	R	Indicates if there are outstanding writes for AXI/CHI port 3. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_3==1 **Programming Mode:** Dynamic
18	wr_port_busy_2	R	Indicates if there are outstanding writes for AXI/CHI port 2. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_2==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
17	wr_port_busy_1	R	Indicates if there are outstanding writes for AXI/CHI port 1. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_1==1 **Programming Mode:** Dynamic
16	wr_port_busy_0	R	Indicates if there are outstanding writes for AXI/CHI port 0. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_0==1 **Programming Mode:** Dynamic
15	rd_port_busy_15	R	Indicates if there are outstanding reads for AXI/CHI port 15. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_15==1 **Programming Mode:** Dynamic
14	rd_port_busy_14	R	Indicates if there are outstanding reads for AXI/CHI port 14. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_14==1 **Programming Mode:** Dynamic
13	rd_port_busy_13	R	Indicates if there are outstanding reads for AXI/CHI port 13. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_13==1 **Programming Mode:** Dynamic
12	rd_port_busy_12	R	Indicates if there are outstanding reads for AXI/CHI port 12. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_12==1 **Programming Mode:** Dynamic
11	rd_port_busy_11	R	Indicates if there are outstanding reads for AXI/CHI port 11. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_11==1 **Programming Mode:** Dynamic
10	rd_port_busy_10	R	Indicates if there are outstanding reads for AXI/CHI port 10. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_10==1 **Programming Mode:** Dynamic
9	rd_port_busy_9	R	Indicates if there are outstanding reads for AXI/CHI port 9. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_9==1 **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
8	rd_port_busy_8	R	Indicates if there are outstanding reads for AXI/CHI port 8. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_8==1 **Programming Mode:** Dynamic
7	rd_port_busy_7	R	Indicates if there are outstanding reads for AXI/CHI port 7. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_7==1 **Programming Mode:** Dynamic
6	rd_port_busy_6	R	Indicates if there are outstanding reads for AXI/CHI port 6. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_6==1 **Programming Mode:** Dynamic
5	rd_port_busy_5	R	Indicates if there are outstanding reads for AXI/CHI port 5. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_5==1 **Programming Mode:** Dynamic
4	rd_port_busy_4	R	Indicates if there are outstanding reads for AXI/CHI port 4. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_4==1 **Programming Mode:** Dynamic
3	rd_port_busy_3	R	Indicates if there are outstanding reads for AXI/CHI port 3. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_3==1 **Programming Mode:** Dynamic
2	rd_port_busy_2	R	Indicates if there are outstanding reads for AXI/CHI port 2. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_2==1 **Programming Mode:** Dynamic
1	rd_port_busy_1	R	Indicates if there are outstanding reads for AXI/CHI port 1. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_1==1 **Programming Mode:** Dynamic
0	rd_port_busy_0	R	Indicates if there are outstanding reads for AXI/CHI port 0. **Value After Reset:** 0x0 **Exists:** UMCTL2_PORT_0==1 **Programming Mode:** Dynamic

1.5 REGB_ADDR_MAP0 Registers

1.5.1 ADDRMAP0

- **Description:** Address Map Register 0
- **Size:** 32 bits
- **Offset:** 0x30000
- **Exists:** UMCTL2_DATA_CHANNEL_INTERLEAVE_EN_1==1

This register is in block REGB_ADDR_MAP0.

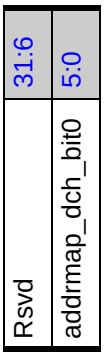


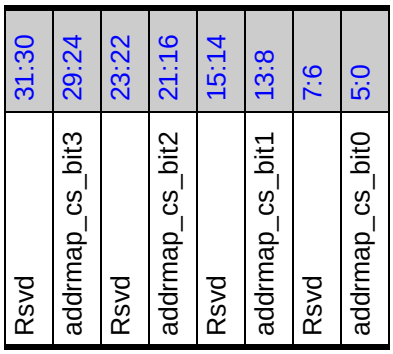
Table 1-271 Fields for Register: ADDRMAP0

Bits	Name	Memory Access	Description
31:6			**Reserved Field:** Yes
5:0	addrmap_dch_bit0	R/W	Selects the HIF address bit used as data channel address bit 0. Valid Range: 0 to 35, and 63 (Traffic constraints apply based on the register value when UMCTL2_EXCL_ACCESS>0. See Exclusive Access section for details.) Internal Base: 3 The selected address bit is determined by adding the internal base to the value of this field. If unused, set to 63 and then channel bit is set to 0. **Value After Reset:** 0x0 **Exists:** UMCTL2_DATA_CHANNEL_INTERLEAVE_EN_1==1 **Programming Mode:** Static

1.5.2 ADDRMAP1

- **Description:** Address Map Register 1
- **Size:** 32 bits
- **Offset:** 0x30004
- **Exists:** MEMC_NUM_RANKS_GT_1==1

This register is in block REGB_ADDR_MAP0.



31:30	29:24	23:22	21:16	15:14	13:8	7:6	5:0
Rsvd	addrmap_cs_bit3	Rsvd	addrmap_cs_bit2	Rsvd	addrmap_cs_bit1	Rsvd	addrmap_cs_bit0

Table 1-272 Fields for Register: ADDRMAP1

Bits	Name	Memory Access	Description
31:30			**Reserved Field:** Yes
29:24	addrmap_cs_bit3	R/W	Selects the HIF address bit used as rank address bit 3. Valid Range: 0 to 26, and 63 Internal Base: 9 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 63 and then rank address bit 3 is set to 0. **Value After Reset:** 0x0 **Exists:** MEMC_NUM_RANKS>8 **Programming Mode:** Static
23:22			**Reserved Field:** Yes
21:16	addrmap_cs_bit2	R/W	Selects the HIF address bit used as rank address bit 2. Valid Range: 0 to 27, and 63 Internal Base: 8 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 63 and then rank address bit 2 is set to 0. **Value After Reset:** 0x0 **Exists:** MEMC_NUM_RANKS>4 **Programming Mode:** Static
15:14			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
13:8	addrmap_cs_bit1	R/W	Selects the HIF address bit used as rank address bit 1. Valid Range: 0 to 32, and 63 Internal Base: 7 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 63 and then rank address bit 1 is set to 0. **Value After Reset:** 0x0 **Exists:** MEMC_NUM_RANKS>2 **Programming Mode:** Static
7:6			**Reserved Field:** Yes
5:0	addrmap_cs_bit0	R/W	Selects the HIF address bit used as rank address bit 0. Valid Range: 0 to 33, and 63 Internal Base: 6 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 63 and then rank address bit 0 is set to 0. **Value After Reset:** 0x0 **Exists:** MEMC_NUM_RANKS>1 **Programming Mode:** Static

1.5.3 ADDRMAP3

- **Description:** Address Map Register 3
- **Size:** 32 bits
- **Offset:** 0x3000c
- **Exists:** Always

This register is in block REGB_ADDR_MAP0.

31:22	21:16	15:14	13:8	7:6	5:0
Rsvd	addrmap_bank_b2	Rsvd	addrmap_bank_b1	Rsvd	addrmap_bank_b0

Table 1-273 Fields for Register: ADDRMAP3

Bits	Name	Memory Access	Description
31:22			**Reserved Field:** Yes
21:16	addrmap_bank_b2	R/W	Selects the HIF address bit used as bank address bit 2. Valid Range: 0 to 34, and 63 Internal Base: 5 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 63 and then bank address bit 2 is set to 0. For LPDDR5 16B mode, this is not used and ADDRMAP4.addrmap_bg_b0 is used as bank address bit 2 instead. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Static
15:14			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
13:8	addrmap_bank_b1	R/W	Selects the HIF address bits used as bank address bit 1. Valid Range: 0 to 35, and 63 Internal Base: 4 The selected HIF address bit for each of the bank address bits is determined by adding the internal base to the value of this field. If unused, set to 63 and then bank address bit 1 is set to 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:6			**Reserved Field:** Yes
5:0	addrmap_bank_b0	R/W	Selects the HIF address bits used as bank address bit 0. Valid Range: 0 to 36, and 63 Internal Base: 3 The selected HIF address bit for each of the bank address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.5.4 ADDRMAP4

- **Description:** Address Map Register 4
- **Size:** 32 bits
- **Offset:** 0x30010
- **Exists:** Always

This register is in block REGB_ADDR_MAP0.

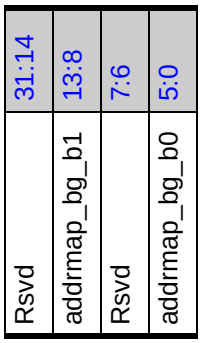


Table 1-274 Fields for Register: ADDRMAP4

Bits	Name	Memory Access	Description
31:14			**Reserved Field:** Yes
13:8	addrmap_bg_b1	R/W	Selects the HIF address bits used as bank group address bit 1. Valid Range: 0 to 35, and 63 Internal Base: 4 The selected HIF address bit for each of the bank group address bits is determined by adding the internal base to the value of this field. If unused, set to 63 and then bank group address bit 1 is set to 0. For LPDDR5 16B mode, this is used as bank address bit 3. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:6			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
5:0	addrmap_bg_b0	R/W	Selects the HIF address bits used as bank group address bit 0. Valid Range: 0 to 36, and 63 Internal Base: 3 The selected HIF address bit for each of the bank group address bits is determined by adding the internal base to the value of this field. If unused, set to 63 and then bank group address bit 0 is set to 0. For LPDDR5 16B mode, this is used as bank address bit 2. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.5.5 ADDRMAP5

- **Description:** Address Map Register 5
- **Size:** 32 bits
- **Offset:** 0x30014
- **Exists:** Always

This register is in block REGB_ADDR_MAP0.

31:29	28:24	23:21	20:16	15:13	12:8	7:5	4:0
Rsvd	addrmap_col_b10	Rsvd	addrmap_col_b9	Rsvd	addrmap_col_b8	Rsvd	addrmap_col_b7

Table 1-275 Fields for Register: ADDRMAP5

Bits	Name	Memory Access	Description
31:29			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
28:24	addrmap_col_b10	R/W	Selects the HIF address bit used as column address bit 10. Valid Range: 0 to 7, x, and 31. x indicates a valid value in the inline ECC configuration. Internal Base: 10 The selected HIF address bit is determined by adding the internal base to the value of this field. Note: In Inline ECC configuration (MEMC_INLINE_ECC=1) and ECC is enabled (ECCCFG0.ecc_mode>0), the highest 3 column address bits must map to the highest 3 valid HIF address bits. If column bit 10 is the highest column address bit, it must map to the highest valid HIF address bit. (x = the highest valid HIF address bit - internal base) If column bit 10 is the second highest column address bit, it must map to the second highest valid HIF address bit. (x = the highest valid HIF address bit - 1 - internal base) If column bit 10 is the third highest column address bit, it must map to the third highest valid HIF address bit. (x = the highest valid HIF address bit - 2 - internal base) If unused, set to 31 and then this column address bit is set to 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
23:21			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
20:16	addrmap_col_b9	R/W	Selects the HIF address bit used as column address bit 9. Valid Range: 0 to 7, x, and 31. x indicates a valid value in the inline ECC configuration. Internal Base: 9 The selected HIF address bit is determined by adding the internal base to the value of this field. Note: In Inline ECC configuration (MEMC_INLINE_ECC=1) and ECC is enabled (ECCCFG0.ecc_mode>0), the highest 3 column address bits must map to the highest 3 valid HIF address bits. If column bit 9 is the highest column address bit, it must map to the highest valid HIF address bit. (x = the highest valid HIF address bit - internal base) If column bit 9 is the second highest column address bit, it must map to the second highest valid HIF address bit. (x = the highest valid HIF address bit - 1 - internal base) If column bit 9 is the third highest column address bit, it must map to the third highest valid HIF address bit. (x = the highest valid HIF address bit - 2 - internal base) If unused, set to 31 and then this column address bit is set to 0. For LPDDR5, this is used as column address bit 5. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
15:13			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
12:8	addrmap_col_b8	R/W	Selects the HIF address bit used as column address bit 8. Valid Range: 0 to 7, x, and 31. x indicates a valid value in the inline ECC configuration. Internal Base: 8 The selected HIF address bit is determined by adding the internal base to the value of this field. Note: In Inline ECC configuration (MEMC_INLINE_ECC=1) and ECC is enabled (ECCCFG0.ecc_mode>0), the highest 3 column address bits must map to the highest 3 valid HIF address bits. If column bit 8 is the second highest column address bit, it must map to the second highest valid HIF address bit. (x = the highest valid HIF address bit - 1 - internal base) If column bit 8 is the third highest column address bit, it must map to the third highest valid HIF address bit. (x = the highest valid HIF address bit - 2 - internal base) If unused, set to 31 and then this column address bit is set to 0. For LPDDR5, this is used as column address bit 4. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:5			**Reserved Field:** Yes
4:0	addrmap_col_b7	R/W	Selects the HIF address bit used as column address bit 7. Valid Range: 0 to 7, x, and 31. x indicates a valid value in the inline ECC configuration. Internal Base: 7 The selected HIF address bit is determined by adding the internal base to the value of this field. In Inline ECC configuration (MEMC_INLINE_ECC=1) and ECC is enabled (ECCCFG0.ecc_mode>0), the highest 3 column address bits must map to the highest 3 valid HIF address bits. If column bit 7 is the third highest column address bit, it must map to the third highest valid HIF address bit. (x = the highest valid HIF address bit - 2 - internal base) If unused, set to 31 and then this column address bit is set to 0. For LPDDR5, this is used as column address bit 3. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.5.6 ADDRMAP6

- **Description:** Address Map Register 6
- **Size:** 32 bits
- **Offset:** 0x30018
- **Exists:** Always

This register is in block REGB_ADDR_MAP0.

31:28	27:24	23:20	19:16	15:12	11:8	7:4	3:0
Rsvd	addrmap_col_b6	Rsvd	addrmap_col_b5	Rsvd	addrmap_col_b4	Rsvd	addrmap_col_b3

Table 1-276 Fields for Register: ADDRMAP6

Bits	Name	Memory Access	Description
31:28			**Reserved Field:** Yes
27:24	addrmap_col_b6	R/W	Selects the HIF address bit used as column address bit 6. Valid Range: 0 to 7, x and 15. x indicates a valid value in the inline ECC configuration. Internal Base: 6 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 15 and then this column address bit is set to 0. For LPDDR5, this is used as column address bit 2. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
23:20			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
19:16	addrmap_col_b5	R/W	Selects the HIF address bit used as column address bit 5. Valid Range: 0 to 7, and 15 Internal Base: 5 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 15 and then this column address bit is set to 0. For LPDDR5, this is used as column address bit 1. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
15:12			**Reserved Field:** Yes
11:8	addrmap_col_b4	R/W	Selects the HIF address bit used as column address bit 4. Valid Range: 0 to 7, and 15 Internal Base: 4 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 15 and then this column address bit is set to 0. For LPDDR5, this is used as column address bit 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:4			**Reserved Field:** Yes
3:0	addrmap_col_b3	R/W	Selects the HIF address bit used as column address bit 3. Valid Range: 0 to 7. Internal Base: 3 The selected HIF address bit is determined by adding the internal base to the value of this field. For LPDDR5, this is used as burst address bit 3. Note : In LPDDR4/5 or DDR5, it is required to program this to 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.5.7 ADDRMAP7

- **Description:** Address Map Register 7
- **Size:** 32 bits
- **Offset:** 0x3001c
- **Exists:** Always

This register is in block REGB_ADDR_MAP0.

31:29	28:24	23:21	20:16	15:13	12:8	7:5	4:0
Rsvd	addrmap_row_b17	Rsvd	addrmap_row_b16	Rsvd	addrmap_row_b15	Rsvd	addrmap_row_b14

Table 1-277 Fields for Register: ADDRMAP7

Bits	Name	Memory Access	Description
31:29			**Reserved Field:** Yes
28:24	addrmap_row_b17	R/W	Selects the HIF address bit used as row address bit 17. Valid Range: 0 to 16, and 31 Internal Base: 23 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then row address bit 17 is set to 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
23:21			**Reserved Field:** Yes
20:16	addrmap_row_b16	R/W	Selects the HIF address bit used as row address bit 16. Valid Range: 0 to 16, and 31 Internal Base: 22 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then row address bit 16 is set to 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
15:13			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
12:8	addrmap_row_b15	R/W	Selects the HIF address bit used as row address bit 15. Valid Range: 0 to 16, and 31 Internal Base: 21 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then row address bit 15 is set to 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:5			**Reserved Field:** Yes
4:0	addrmap_row_b14	R/W	Selects the HIF address bit used as row address bit 14. Valid Range: 0 to 16, and 31 Internal Base: 20 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then row address bit 14 is set to 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.5.8 ADDRMAP8

- **Description:** Address Map Register 8
- **Size:** 32 bits
- **Offset:** 0x30020
- **Exists:** Always

This register is in block REGB_ADDR_MAP0.

31:29	28:24	23:21	20:16	15:13	12:8	7:5	4:0
Rsvd	addrmap_row_b13	Rsvd	addrmap_row_b12	Rsvd	addrmap_row_b11	Rsvd	addrmap_row_b10

Table 1-278 Fields for Register: ADDRMAP8

Bits	Name	Memory Access	Description
31:29			**Reserved Field:** Yes
28:24	addrmap_row_b13	R/W	Selects the HIF address bit used as row address bit 13. Valid Range: 0 to 16, and 31 Internal Base: 19 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then row address bit 13 is set to 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
23:21			**Reserved Field:** Yes
20:16	addrmap_row_b12	R/W	Selects the HIF address bit used as row address bit 12. Valid Range: 0 to 16, and 31 Internal Base: 18 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then row address bit 12 is set to 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
15:13			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
12:8	addrmap_row_b11	R/W	Selects the HIF address bit used as row address bit 11. Valid Range: 0 to 16 Internal Base: 17 The selected HIF address bit is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:5			**Reserved Field:** Yes
4:0	addrmap_row_b10	R/W	Selects the HIF address bits used as row address bit 10. Valid Range: 0 to 16 Internal Base: 16 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.5.9 ADDRMAP9

- **Description:** Address Map Register 9
- **Size:** 32 bits
- **Offset:** 0x30024
- **Exists:** Always

This register is in block REGB_ADDR_MAP0.

31:29	28:24	23:21	20:16	15:13	12:8	7:5	4:0
Rsvd	addrmap_row_b9	Rsvd	addrmap_row_b8	Rsvd	addrmap_row_b7	Rsvd	addrmap_row_b6

Table 1-279 Fields for Register: ADDRMAP9

Bits	Name	Memory Access	Description
31:29			**Reserved Field:** Yes
28:24	addrmap_row_b9	R/W	Selects the HIF address bits used as row address bit 9. Valid Range: 0 to 16 Internal Base: 15 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
23:21			**Reserved Field:** Yes
20:16	addrmap_row_b8	R/W	Selects the HIF address bits used as row address bit 8. Valid Range: 0 to 16 Internal Base: 14 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
15:13			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
12:8	addrmap_row_b7	R/W	Selects the HIF address bits used as row address bit 7. Valid Range: 0 to 16 Internal Base: 13 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:5			**Reserved Field:** Yes
4:0	addrmap_row_b6	R/W	Selects the HIF address bits used as row address bit 6. Valid Range: 0 to 16 Internal Base: 12 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.5.10 ADDRMAP10

- **Description:** Address Map Register 10
- **Size:** 32 bits
- **Offset:** 0x30028
- **Exists:** Always

This register is in block REGB_ADDR_MAP0.

31:29	28:24	23:21	20:16	15:13	12:8	7:5	4:0
Rsvd	addrmap_row_b5	Rsvd	addrmap_row_b4	Rsvd	addrmap_row_b3	Rsvd	addrmap_row_b2

Table 1-280 Fields for Register: ADDRMAP10

Bits	Name	Memory Access	Description
31:29			**Reserved Field:** Yes
28:24	addrmap_row_b5	R/W	Selects the HIF address bits used as row address bit 5. Valid Range: 0 to 16 Internal Base: 11 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
23:21			**Reserved Field:** Yes
20:16	addrmap_row_b4	R/W	Selects the HIF address bits used as row address bit 4. Valid Range: 0 to 16 Internal Base: 10 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
15:13			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
12:8	addrmap_row_b3	R/W	Selects the HIF address bits used as row address bit 3. Valid Range: 0 to 16 Internal Base: 9 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:5			**Reserved Field:** Yes
4:0	addrmap_row_b2	R/W	Selects the HIF address bits used as row address bit 2. Valid Range: 0 to 16 Internal Base: 8 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.5.11 ADDRMAP11

- **Description:** Address Map Register 11
- **Size:** 32 bits
- **Offset:** 0x3002c
- **Exists:** Always

This register is in block REGB_ADDR_MAP0.

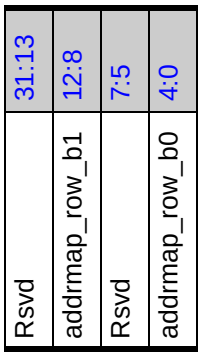


Table 1-281 Fields for Register: ADDRMAP11

Bits	Name	Memory Access	Description
31:13			**Reserved Field:** Yes
12:8	addrmap_row_b1	R/W	Selects the HIF address bits used as row address bit 1. Valid Range: 0 to 16 Internal Base: 7 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:5			**Reserved Field:** Yes
4:0	addrmap_row_b0	R/W	Selects the HIF address bits used as row address bit 0. Valid Range: 0 to 16 Internal Base: 6 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.5.12 ADDRMAP12

- **Description:** Address Map Register 12
- **Size:** 32 bits
- **Offset:** 0x30030
- **Exists:** Always

This register is in block REGB_ADDR_MAP0.

Table 1-282 Fields for Register: ADDRMAP12

Bits	Name	Memory Access	Description
31:3			**Reserved Field:** Yes
2:0	nonbinary_device_density	R/W	Indicates what type of SDRAM device is in use. ■ 3'b000: All addresses are valid ■ 3'b001: Every address having row[13:12]==2'b11 is considered as invalid ■ 3'b010: Every address having row[14:13]==2'b11 is considered as invalid ■ 3'b011: Every address having row[15:14]==2'b11 is considered as invalid ■ 3'b100: Every address having row[16:15]==2'b11 is considered as invalid ■ 3'b101: Every address having row[17:16]==2'b11 is considered as invalid **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.5.13 ADDRMAPLUTCFG

- **Description:** Addr CS map LUT config register
- **Size:** 32 bits
- **Offset:** 0x30080
- **Exists:** DDRCTL_LUT_ADDRMAP==1

This register is in block REGB_ADDR_MAP0.

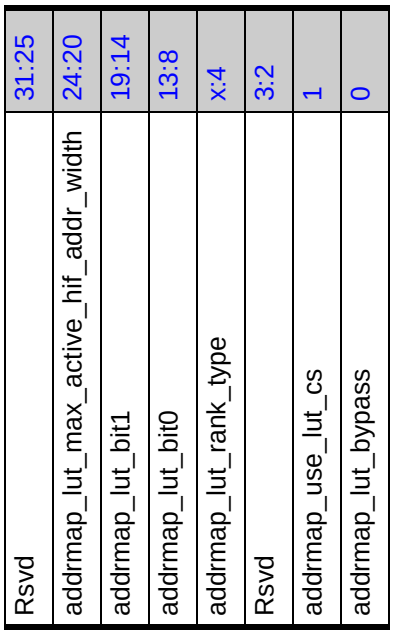


Table 1-283 Fields for Register: ADDRMAPLUTCFG

Bits	Name	Memory Access	Description
31:25			**Reserved Field:** Yes
24:20	addrmap_lut_max_active_hif_addr_width	R/W	This register specifies the maximum active width of HIF address. The condition (addrmap_lut_max_active_hif_addr_width + base) <= `MEMC_HIF_ADDR_WIDTH must be met. Internal Base: 20 **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

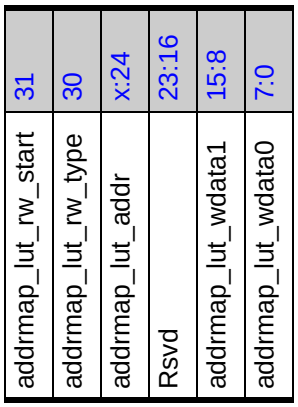
Bits	Name	Memory Access	Description
19:14	addrmap_lut_bit1	R/W	The bit selection offset of the highest bit of LUT index. Valid Range: addrmap_lut_bit0 to (addrmap_lut_max_active_hif_addr_width + 20 - DDRCTL_LUT_CS_WIN_BITS + 1 - base), and 63 Internal Base: 7 The bit selection offset bit1 is determined by adding the internal base to the value of this field. If unused, set to 63. Note: 1. Addrmap_lut_bit1 can only be used when addrmap_lut_bit0 is used. 2. When addrmap_lut_bit1 and addrmap_lut_bit0 are used, addrmap_lut_bit1 must be equal or greater than addrmap_lut_bit0. **Value After Reset:** 0x3f **Exists:** Always **Programming Mode:** Static
13:8	addrmap_lut_bit0	R/W	When addrmap_lut_bit1 is not used, it specifies the bit selection offset of the highest bit of LUT index. When addrmap_lut_bit1 is used, it specifies the bit selection offset of the 2nd highest bit of LUT index. Valid Range: 0 to (addrmap_lut_max_active_hif_addr_width + 20 - DDRCTL_LUT_CS_WIN_BITS - base), and 63 Internal Base: 6 The bit selection offset bit0 is determined by adding the internal base to the value of this field. If unused, set to 63. **Value After Reset:** 0x3f **Exists:** Always **Programming Mode:** Static
x:4	addrmap_lut_rank_type	R/W	Select one mapping from the two settings of second level mapper. Each bit corresponds to each rank. ■ 0: Uses address mapping 0 ■ 1: Uses address mapping 1 **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static **Range Variable[x]:** "MEMC_NUM_RANKS" + 3
3:2			**Reserved Field:** Yes
1	addrmap_use_lut_cs	R/W	Set it to 1 to use the CS Map LUT design to get corresponding CS. Assert only if addr_map_lut_bypass = 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

Bits	Name	Memory Access	Description
0	addrmap_lut_bypass	R/W	Set it to 0 to use the remapped address from LUT in second level mapper. Set it to 1 to bypass the output of LUT. Only the legacy address mapping will take effects on the mapping from HIF address to DDR physical address. **Value After Reset:** 0x1 **Exists:** Always **Programming Mode:** Static

1.5.14 ADDRMAPLUTCTRL

- **Description:** Addr CS map LUT control register
- **Size:** 32 bits
- **Offset:** 0x30084
- **Exists:** DDRCTL_LUT_ADDRMAP==1

This register is in block REGB_ADDR_MAP0.



31	30	x:24	23:16	15:8	7:0
addrmap_lut_rw_start	addrmap_lut_rw_type	addrmap_lut_addr	Rsvd	addrmap_lut_wdata1	addrmap_lut_wdata0

Table 1-284 Fields for Register: ADDRMAPLUTCTRL

Bits	Name	Memory Access	Description
31	addrmap_lut_rw_start	R/W1C	Set this bit to 1 to trigger a LUT Read/Write Operation. This bit is self-cleared. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
30	addrmap_lut_rw_type	R/W	LUT read/write operation type. ■ 0: Indicate a read operation. The return data from LUT[ADDR] will be loaded into ADDRMAPLUTRDATA register. ■ 1: Indicates a write operation. The data of addrmap_lut_wdata1/addrmap_lut_wdata0 will be written into LUT[ADDR]. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

Bits	Name	Memory Access	Description
x:24	addrmap_lut_addr	R/W	LUT entry Read/Write address. Notes: 1. For read operation, the ADDR shall be even. Two LUT entries will be loaded into ADDRMAP_LUT_RDATA registers 2. For write operation, the ADDR shall be even. Two entries will be written into the LUT **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic **Range Variable[x]:** "DDRCTL_LUT_ADDRMAP_CS_WIN_BITS" + 23
23:16			**Reserved Field:** Yes
15:8	addrmap_lut_wdata1	R/W	8 bits entry data to be written to LUT[addrmap_lut_addr+1]. ■ [7]: Entry Valid. 1 - valid, 0 - invalid. ■ [6:5]: CSn[1:0], mapped corresponding CS for LUT output. ■ [DDRCTL_LUT_CS_WIN_BITS-1:0]: Mapped_addr[4:0], mapped address **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic
7:0	addrmap_lut_wdata0	R/W	8 bits entry data to be written to LUT[addrmap_lut_addr]. ■ [7]: Entry Valid. 1 indicates that entry is used for LUT address mapping. ■ [6:5]: CSn[1:0], mapped corresponding CS for LUT output. ■ [DDRCTL_LUT_CS_WIN_BITS-1:0]: The most significant bits in the remapping address for one corresponding rank. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Dynamic

1.5.15 ADDRMAPLUTRDATA

- **Description:** Addr CS map LUT read data register
- **Size:** 32 bits
- **Offset:** 0x30088
- **Exists:** DDRCTL_LUT_ADDRMAP==1

This register is in block REGB_ADDR_MAP0.

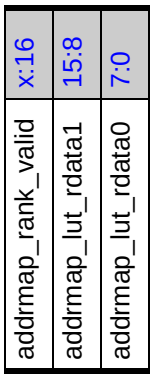


Table 1-285 Fields for Register: ADDRMAPLUTRDATA

Bits	Name	Memory Access	Description
x:16	addrmap_rank_valid	R	Present rank valid value based on rank status. Each bit corresponds each rank. **Value After Reset:** 0x0 **Exists:** DDRCTL_LUT_ADDRMAP==1 **Programming Mode:** Static **Range Variable[x]:** "MEMC_NUM_RANKS" + 15
15:8	addrmap_lut_rdata1	R	The read-only register to store LUT[ADDRMAPLUTCTRL.addrmap_lut_addr+1] of previous LUT read operation. **Value After Reset:** 0x0 **Exists:** DDRCTL_LUT_ADDRMAP==1 **Programming Mode:** Static
7:0	addrmap_lut_rdata0	R	The read-only register to store LUT[ADDRMAPLUTCTRL.addrmap_lut_addr] of previous LUT read operation. **Value After Reset:** 0x0 **Exists:** DDRCTL_LUT_ADDRMAP==1 **Programming Mode:** Static

1.6 REGB_ADDR_MAP1 Registers

1.6.1 ADDRMAP1

- **Description:** Address Map Register 1
- **Size:** 32 bits
- **Offset:** 0x31004
- **Exists:** MEMC_NUM_RANKS_GT_1==1

This register is in block REGB_ADDR_MAP1.

31:30	29:24	23:22	21:16	15:14	13:8	7:6	5:0
Rsvd	addrmap_cs_bit3	Rsvd	addrmap_cs_bit2	Rsvd	addrmap_cs_bit1	Rsvd	addrmap_cs_bit0

Table 1-286 Fields for Register: ADDRMAP1

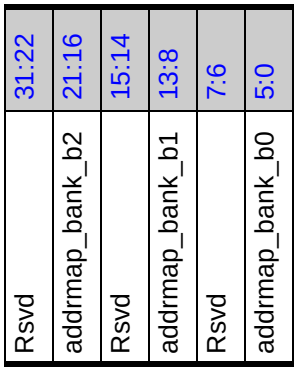
Bits	Name	Memory Access	Description
31:30			**Reserved Field:** Yes
29:24	addrmap_cs_bit3	R/W	Selects the HIF address bit used as rank address bit 3. Valid Range: 0 to 26, and 63 Internal Base: 9 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 63 and then rank address bit 3 is set to 0. **Value After Reset:** 0x0 **Exists:** MEMC_NUM_RANKS>8 **Programming Mode:** Static
23:22			**Reserved Field:** Yes
21:16	addrmap_cs_bit2	R/W	Selects the HIF address bit used as rank address bit 2. Valid Range: 0 to 27, and 63 Internal Base: 8 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 63 and then rank address bit 2 is set to 0. **Value After Reset:** 0x0 **Exists:** MEMC_NUM_RANKS>4 **Programming Mode:** Static
15:14			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
13:8	addrmap_cs_bit1	R/W	Selects the HIF address bit used as rank address bit 1. Valid Range: 0 to 32, and 63 Internal Base: 7 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 63 and then rank address bit 1 is set to 0. **Value After Reset:** 0x0 **Exists:** MEMC_NUM_RANKS>2 **Programming Mode:** Static
7:6			**Reserved Field:** Yes
5:0	addrmap_cs_bit0	R/W	Selects the HIF address bit used as rank address bit 0. Valid Range: 0 to 33, and 63 Internal Base: 6 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 63 and then rank address bit 0 is set to 0. **Value After Reset:** 0x0 **Exists:** MEMC_NUM_RANKS>1 **Programming Mode:** Static

1.6.2 ADDRMAP3

- **Description:** Address Map Register 3
- **Size:** 32 bits
- **Offset:** 0x3100c
- **Exists:** Always

This register is in block REGB_ADDR_MAP1.



31:22	21:16	15:14	13:8	7:6	5:0
Rsvd	addrmap_bank_b2	Rsvd	addrmap_bank_b1	Rsvd	addrmap_bank_b0

Table 1-287 Fields for Register: ADDRMAP3

Bits	Name	Memory Access	Description
31:22			**Reserved Field:** Yes
21:16	addrmap_bank_b2	R/W	Selects the HIF address bit used as bank address bit 2. Valid Range: 0 to 34, and 63 Internal Base: 5 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 63 and then bank address bit 2 is set to 0. For LPDDR5 16B mode, this is not used and ADDRMAP4.addrmap_bg_b0 is used as bank address bit 2 instead. **Value After Reset:** 0x0 **Exists:** DDRCTL_LPDDR==1 **Programming Mode:** Static
15:14			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
13:8	addrmap_bank_b1	R/W	Selects the HIF address bits used as bank address bit 1. Valid Range: 0 to 35, and 63 Internal Base: 4 The selected HIF address bit for each of the bank address bits is determined by adding the internal base to the value of this field. If unused, set to 63 and then bank address bit 1 is set to 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:6			**Reserved Field:** Yes
5:0	addrmap_bank_b0	R/W	Selects the HIF address bits used as bank address bit 0. Valid Range: 0 to 36, and 63 Internal Base: 3 The selected HIF address bit for each of the bank address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.6.3 ADDRMAP4

- **Description:** Address Map Register 4
- **Size:** 32 bits
- **Offset:** 0x31010
- **Exists:** Always

This register is in block REGB_ADDR_MAP1.

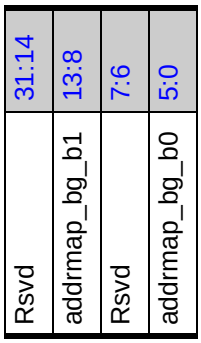


Table 1-288 Fields for Register: ADDRMAP4

Bits	Name	Memory Access	Description
31:14			**Reserved Field:** Yes
13:8	addrmap_bg_b1	R/W	Selects the HIF address bits used as bank group address bit 1. Valid Range: 0 to 35, and 63 Internal Base: 4 The selected HIF address bit for each of the bank group address bits is determined by adding the internal base to the value of this field. If unused, set to 63 and then bank group address bit 1 is set to 0. For LPDDR5 16B mode, this is used as bank address bit 3. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:6			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
5:0	addrmap_bg_b0	R/W	Selects the HIF address bits used as bank group address bit 0. Valid Range: 0 to 36, and 63 Internal Base: 3 The selected HIF address bit for each of the bank group address bits is determined by adding the internal base to the value of this field. If unused, set to 63 and then bank group address bit 0 is set to 0. For LPDDR5 16B mode, this is used as bank address bit 2. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.6.4 ADDRMAP5

- **Description:** Address Map Register 5
- **Size:** 32 bits
- **Offset:** 0x31014
- **Exists:** Always

This register is in block REGB_ADDR_MAP1.

31:29	28:24	23:21	20:16	15:13	12:8	7:5	4:0
Rsvd	addrmap_col_b10	Rsvd	addrmap_col_b9	Rsvd	addrmap_col_b8	Rsvd	addrmap_col_b7

Table 1-289 Fields for Register: ADDRMAP5

Bits	Name	Memory Access	Description
31:29			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
28:24	addrmap_col_b10	R/W	Selects the HIF address bit used as column address bit 10. Valid Range: 0 to 7, x, and 31. x indicates a valid value in the inline ECC configuration. Internal Base: 10 The selected HIF address bit is determined by adding the internal base to the value of this field. Note: In Inline ECC configuration (MEMC_INLINE_ECC=1) and ECC is enabled (ECCCFG0.ecc_mode>0), the highest 3 column address bits must map to the highest 3 valid HIF address bits. If column bit 10 is the highest column address bit, it must map to the highest valid HIF address bit. (x = the highest valid HIF address bit - internal base) If column bit 10 is the second highest column address bit, it must map to the second highest valid HIF address bit. (x = the highest valid HIF address bit - 1 - internal base) If column bit 10 is the third highest column address bit, it must map to the third highest valid HIF address bit. (x = the highest valid HIF address bit - 2 - internal base) If unused, set to 31 and then this column address bit is set to 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
23:21			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
20:16	addrmap_col_b9	R/W	Selects the HIF address bit used as column address bit 9. Valid Range: 0 to 7, x, and 31. x indicates a valid value in the inline ECC configuration. Internal Base: 9 The selected HIF address bit is determined by adding the internal base to the value of this field. Note: In Inline ECC configuration (MEMC_INLINE_ECC=1) and ECC is enabled (ECCCFG0.ecc_mode>0), the highest 3 column address bits must map to the highest 3 valid HIF address bits. If column bit 9 is the highest column address bit, it must map to the highest valid HIF address bit. (x = the highest valid HIF address bit - internal base) If column bit 9 is the second highest column address bit, it must map to the second highest valid HIF address bit. (x = the highest valid HIF address bit - 1 - internal base) If column bit 9 is the third highest column address bit, it must map to the third highest valid HIF address bit. (x = the highest valid HIF address bit - 2 - internal base) If unused, set to 31 and then this column address bit is set to 0. For LPDDR5, this is used as column address bit 5. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
15:13			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
12:8	addrmap_col_b8	R/W	Selects the HIF address bit used as column address bit 8. Valid Range: 0 to 7, x, and 31. x indicates a valid value in the inline ECC configuration. Internal Base: 8 The selected HIF address bit is determined by adding the internal base to the value of this field. Note: In Inline ECC configuration (MEMC_INLINE_ECC=1) and ECC is enabled (ECCCFG0.ecc_mode>0), the highest 3 column address bits must map to the highest 3 valid HIF address bits. If column bit 8 is the second highest column address bit, it must map to the second highest valid HIF address bit. (x = the highest valid HIF address bit - 1 - internal base) If column bit 8 is the third highest column address bit, it must map to the third highest valid HIF address bit. (x = the highest valid HIF address bit - 2 - internal base) If unused, set to 31 and then this column address bit is set to 0. For LPDDR5, this is used as column address bit 4. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:5			**Reserved Field:** Yes
4:0	addrmap_col_b7	R/W	Selects the HIF address bit used as column address bit 7. Valid Range: 0 to 7, x, and 31. x indicates a valid value in the inline ECC configuration. Internal Base: 7 The selected HIF address bit is determined by adding the internal base to the value of this field. In Inline ECC configuration (MEMC_INLINE_ECC=1) and ECC is enabled (ECCCFG0.ecc_mode>0), the highest 3 column address bits must map to the highest 3 valid HIF address bits. If column bit 7 is the third highest column address bit, it must map to the third highest valid HIF address bit. (x = the highest valid HIF address bit - 2 - internal base) If unused, set to 31 and then this column address bit is set to 0. For LPDDR5, this is used as column address bit 3. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.6.5 ADDRMAP6

- **Description:** Address Map Register 6
- **Size:** 32 bits
- **Offset:** 0x31018
- **Exists:** Always

This register is in block REGB_ADDR_MAP1.

31:28	27:24	23:20	19:16	15:12	11:8	7:4	3:0
Rsvd	addrmap_col_b6	Rsvd	addrmap_col_b5	Rsvd	addrmap_col_b4	Rsvd	addrmap_col_b3

Table 1-290 Fields for Register: ADDRMAP6

Bits	Name	Memory Access	Description
31:28			**Reserved Field:** Yes
27:24	addrmap_col_b6	R/W	Selects the HIF address bit used as column address bit 6. Valid Range: 0 to 7, x and 15. x indicates a valid value in the inline ECC configuration. Internal Base: 6 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 15 and then this column address bit is set to 0. For LPDDR5, this is used as column address bit 2. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
23:20			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
19:16	addrmap_col_b5	R/W	Selects the HIF address bit used as column address bit 5. Valid Range: 0 to 7, and 15 Internal Base: 5 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 15 and then this column address bit is set to 0. For LPDDR5, this is used as column address bit 1. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
15:12			**Reserved Field:** Yes
11:8	addrmap_col_b4	R/W	Selects the HIF address bit used as column address bit 4. Valid Range: 0 to 7, and 15 Internal Base: 4 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 15 and then this column address bit is set to 0. For LPDDR5, this is used as column address bit 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:4			**Reserved Field:** Yes
3:0	addrmap_col_b3	R/W	Selects the HIF address bit used as column address bit 3. Valid Range: 0 to 7. Internal Base: 3 The selected HIF address bit is determined by adding the internal base to the value of this field. For LPDDR5, this is used as burst address bit 3. Note : In LPDDR4/5 or DDR5, it is required to program this to 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.6.6 ADDRMAP7

- **Description:** Address Map Register 7
- **Size:** 32 bits
- **Offset:** 0x3101c
- **Exists:** Always

This register is in block REGB_ADDR_MAP1.

31:29	28:24	23:21	20:16	15:13	12:8	7:5	4:0
Rsvd	addrmap_row_b17	Rsvd	addrmap_row_b16	Rsvd	addrmap_row_b15	Rsvd	addrmap_row_b14

Table 1-291 Fields for Register: ADDRMAP7

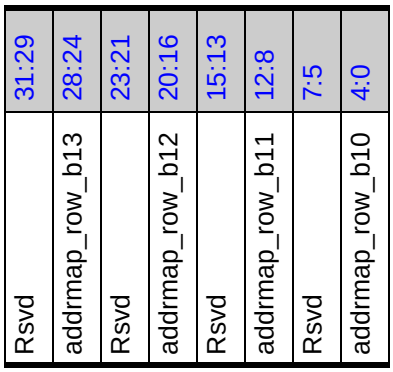
Bits	Name	Memory Access	Description
31:29			**Reserved Field:** Yes
28:24	addrmap_row_b17	R/W	Selects the HIF address bit used as row address bit 17. Valid Range: 0 to 16, and 31 Internal Base: 23 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then row address bit 17 is set to 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
23:21			**Reserved Field:** Yes
20:16	addrmap_row_b16	R/W	Selects the HIF address bit used as row address bit 16. Valid Range: 0 to 16, and 31 Internal Base: 22 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then row address bit 16 is set to 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
15:13			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
12:8	addrmap_row_b15	R/W	Selects the HIF address bit used as row address bit 15. Valid Range: 0 to 16, and 31 Internal Base: 21 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then row address bit 15 is set to 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:5			**Reserved Field:** Yes
4:0	addrmap_row_b14	R/W	Selects the HIF address bit used as row address bit 14. Valid Range: 0 to 16, and 31 Internal Base: 20 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then row address bit 14 is set to 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.6.7 ADDRMAP8

- **Description:** Address Map Register 8
- **Size:** 32 bits
- **Offset:** 0x31020
- **Exists:** Always

This register is in block REGB_ADDR_MAP1.



31:29	28:24	23:21	20:16	15:13	12:8	7:5	4:0
Rsvd	addrmap_row_b13	Rsvd	addrmap_row_b12	Rsvd	addrmap_row_b11	Rsvd	addrmap_row_b10

Table 1-292 Fields for Register: ADDRMAP8

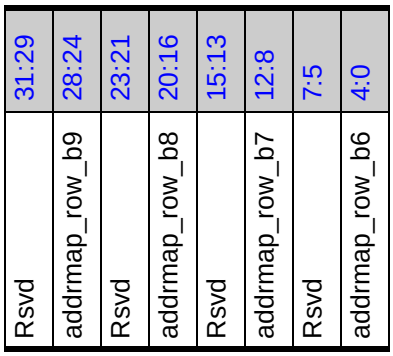
Bits	Name	Memory Access	Description
31:29			**Reserved Field:** Yes
28:24	addrmap_row_b13	R/W	Selects the HIF address bit used as row address bit 13. Valid Range: 0 to 16, and 31 Internal Base: 19 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then row address bit 13 is set to 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
23:21			**Reserved Field:** Yes
20:16	addrmap_row_b12	R/W	Selects the HIF address bit used as row address bit 12. Valid Range: 0 to 16, and 31 Internal Base: 18 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then row address bit 12 is set to 0. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
15:13			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
12:8	addrmap_row_b11	R/W	Selects the HIF address bit used as row address bit 11. Valid Range: 0 to 16 Internal Base: 17 The selected HIF address bit is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:5			**Reserved Field:** Yes
4:0	addrmap_row_b10	R/W	Selects the HIF address bits used as row address bit 10. Valid Range: 0 to 16 Internal Base: 16 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.6.8 ADDRMAP9

- **Description:** Address Map Register 9
- **Size:** 32 bits
- **Offset:** 0x31024
- **Exists:** Always

This register is in block REGB_ADDR_MAP1.



31:29	28:24	23:21	20:16	15:13	12:8	7:5	4:0
Rsvd	addrmap_row_b9	Rsvd	addrmap_row_b8	Rsvd	addrmap_row_b7	Rsvd	addrmap_row_b6

Table 1-293 Fields for Register: ADDRMAP9

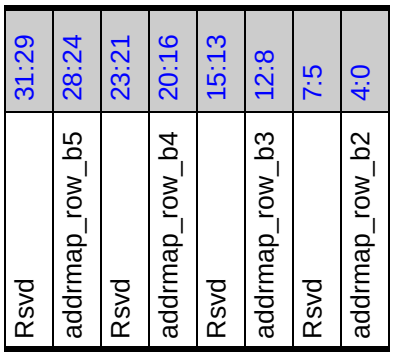
Bits	Name	Memory Access	Description
31:29			**Reserved Field:** Yes
28:24	addrmap_row_b9	R/W	Selects the HIF address bits used as row address bit 9. Valid Range: 0 to 16 Internal Base: 15 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
23:21			**Reserved Field:** Yes
20:16	addrmap_row_b8	R/W	Selects the HIF address bits used as row address bit 8. Valid Range: 0 to 16 Internal Base: 14 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
15:13			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
12:8	addrmap_row_b7	R/W	Selects the HIF address bits used as row address bit 7. Valid Range: 0 to 16 Internal Base: 13 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:5			**Reserved Field:** Yes
4:0	addrmap_row_b6	R/W	Selects the HIF address bits used as row address bit 6. Valid Range: 0 to 16 Internal Base: 12 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.6.9 ADDRMAP10

- **Description:** Address Map Register 10
- **Size:** 32 bits
- **Offset:** 0x31028
- **Exists:** Always

This register is in block REGB_ADDR_MAP1.



31:29	28:24	23:21	20:16	15:13	12:8	7:5	4:0
Rsvd	addrmap_row_b5	Rsvd	addrmap_row_b4	Rsvd	addrmap_row_b3	Rsvd	addrmap_row_b2

Table 1-294 Fields for Register: ADDRMAP10

Bits	Name	Memory Access	Description
31:29			**Reserved Field:** Yes
28:24	addrmap_row_b5	R/W	Selects the HIF address bits used as row address bit 5. Valid Range: 0 to 16 Internal Base: 11 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
23:21			**Reserved Field:** Yes
20:16	addrmap_row_b4	R/W	Selects the HIF address bits used as row address bit 4. Valid Range: 0 to 16 Internal Base: 10 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
15:13			**Reserved Field:** Yes

Bits	Name	Memory Access	Description
12:8	addrmap_row_b3	R/W	Selects the HIF address bits used as row address bit 3. Valid Range: 0 to 16 Internal Base: 9 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:5			**Reserved Field:** Yes
4:0	addrmap_row_b2	R/W	Selects the HIF address bits used as row address bit 2. Valid Range: 0 to 16 Internal Base: 8 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

1.6.10 ADDRMAP11

- **Description:** Address Map Register 11
- **Size:** 32 bits
- **Offset:** 0x3102c
- **Exists:** Always

This register is in block REGB_ADDR_MAP1.

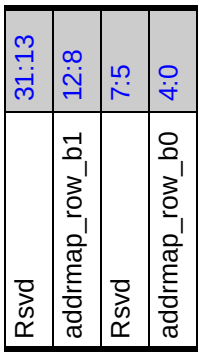


Table 1-295 Fields for Register: ADDRMAP11

Bits	Name	Memory Access	Description
31:13			**Reserved Field:** Yes
12:8	addrmap_row_b1	R/W	Selects the HIF address bits used as row address bit 1. Valid Range: 0 to 16 Internal Base: 7 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static
7:5			**Reserved Field:** Yes
4:0	addrmap_row_b0	R/W	Selects the HIF address bits used as row address bit 0. Valid Range: 0 to 16 Internal Base: 6 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. **Value After Reset:** 0x0 **Exists:** Always **Programming Mode:** Static

2

Programming

This chapter includes the following sections:

2.1 Initialization

This section describes the programming sequence that must be executed at power-up to bring the controller, the PHY, and the memories into a state where HIF reads and writes can be performed.

This section includes the following subsections:

2.1.1 Register Groups

See "Clock And Reset Requirements" in the DesignWare Cores LPDDR5/4/4X Memory Controller Databook for a summary of clocking in DDRCTL.

Some APB registers can be changed at any time. They are defined as dynamic registers. For more information about this, see "Dynamic Registers" on page 668. Other APB registers must not be changed after the `core_ddrc_rstn` is de-asserted. "Group 1: Registers that can be Written when No Read/Write Traffic is Present at the DFI" , "Group 2: Registers that can be Written in Self-refresh Mode" , "Refresh Related Registers" are exceptions to this. These are indicated as Static registers.

The APB registers are reset by the signal `presetn`, which is de-asserted synchronously with `pclk`.

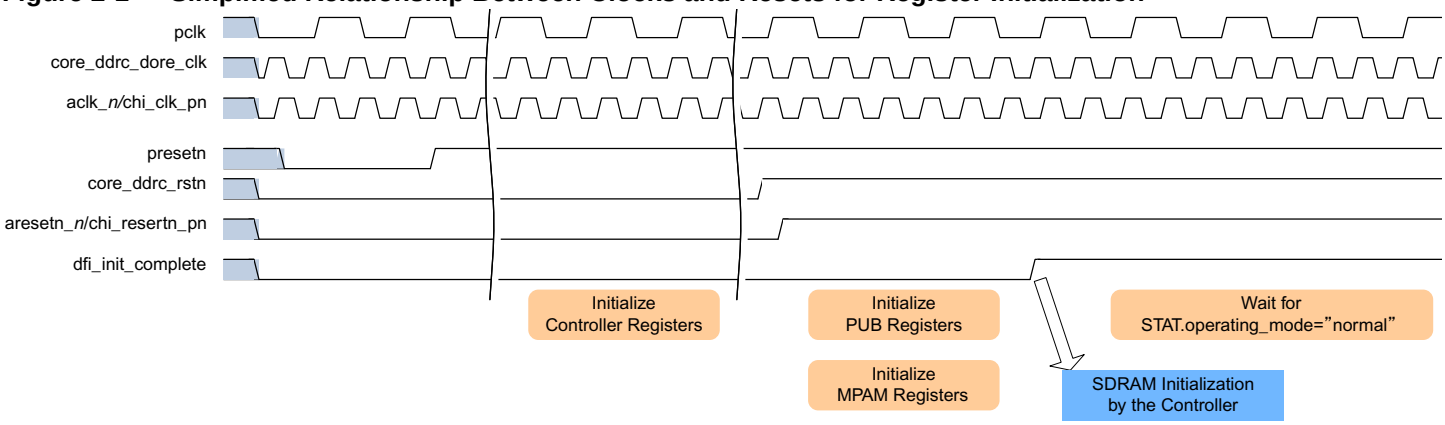
Static registers are sampled directly in the destination domains (`core_ddrc_core_clk/aclk` domains) without synchronizing circuits. The de-assertion of the signal `presetn` is synchronized to the destination domain clocks and used to de-assert the reset on the static registers in their destination domains. Therefore, the static registers must be programmed when `core_ddrc_core_clk/aclk_n` and `pclk` are stable, presetn is de-asserted (at least for 4 cycles of clock in the destination domain) and `core_ddrc_rstn/aresetn_n` are asserted.

Synchronizing circuits are used to transfer the signals for dynamic registers to their destination domains. While these registers must be initialized with `core_ddrc_rstn` asserted, they can be changed thereafter with `core_ddrc_rstn` de-asserted.

2.1.2 Controller Initialization

Figure 2-1 on page 665 shows the relationship between clocks and resets for initialization of registers. Power and clocks must be in accordance with the requirements of full ASIC/SoC design requirements, and the SDRAM power and clocks must in accordance with the SDRAM specifications.

Figure 2-1 Simplified Relationship Between Clocks and Resets for Register Initialization



After power-up:

1. Assert the `core_ddrc_rstn` and `aresetn_n` resets
2. Assert `presetn`
3. Start the clocks (`pclk`, `core_ddrc_core_clk`, `aclk_n`)
4. De-assert `presetn` once the clocks are active and stable
5. Allow 128 APB clock cycles for synchronization of `presetn` to `core_ddrc_core_clk` and `aclk` domains and to permit initialization of end logic
6. Initialize the registers
7. De-assert the resets (`core_ddrc_rstn` and `aresetn_n`)

Any subsequent access to the programming interface of the registers which may be accessed outside of reset must happen when both `pclk` and `core_ddrc_core_clk` are active and stable (see the "Register Descriptions" chapter).

Note

For MPAM enabled configurations:

Since all MPAM registers are implemented in `core_clk` domain, these registers must be accessed only after `core_ddrc_rstn` is de-asserted and `core_clk` is active and stable.

2.1.3 PHY Initialization

Before the memory can be accessed by the DDRCTL controller, the PHY must be initialized.

When using the DWC LPDDR5/4/4X PHY, the PHY Utility Block (PUB) can be used for initialization. The PUB and details on the PHY initialization using the PUB are documented in the DesignWare Cores LPDDR5/4/4X PHY Utility Block (PUB) Databook. For an outline of the full initialization sequence for a DWC LPDDR5/4/4X, refer to Table 2-1 on page 666.

2.1.4 SDRAM Initialization

Synopsys PHY initializes the SDRAM as part of PHY initialization procedure.

2.1.5 LPDDR5/4/4X Initialization Summary Tables

The following tables summarize the initialization procedure when using the Synopsys DWC LPDDR5/4/4X (Table 2-1). For more information about this, see the DesignWare Cores LPDDR5/4/4X PHY Utility Block (PUB) Databook.

Table 2-1 DWC_ddr_umctl5 and Memory Initialization with LPDDR5/4/4X PHY

Step	Application	SVTB Task	Notes
1	Follow the PHYs power up procedure		See PUB databook for details
2	Program the DWC_ddrctl registers		Note 1
3	Disable auto-refreshes, self-refresh, powerdown and assertion of dfi_dram_clk_disable by setting RFSHCTL0.dis_auto_refresh = 1, PWRCTL.powerdown_en = 0 and PWRCTL.selfref_en = 0 PWRTL.en_dfi_dram_clk_disable = 0	reset_dut	
4	De-assert reset signal core_ddrc_rstn		
5	Set SWCTL.sw_done to ’0’	phy_init	
6	Set DFIMISC.dfi_init_complete_en to ’0’	phy_init	
7	Set SWCTL.sw_done to ’1’	phy_init	Require polling SWSTAT.sw_done_ackafter setting SWCTL.sw_done to ’1’
8	Start PHY initialization and training by accessing relevant PUB registers	phy_init	See PUB databook for details
9	Poll the PUB register APBONLY.UctShadowRegs[0] =1’b0	phy_init	See PUB databook for details
10	Read the PUB Register APBONLY.UctWriteOnlyShadow for training status	phy_init	See PUB databook for details
11	Write the PUB Register APBONLY.DctWriteProt = 0	phy_init	See PUB databook for details
12	Poll the PUB register APBONLY.UctShadowRegs[0] =1’b1	phy_init	See PUB databook for details
13	Write the PUB Register APBONLY.DctWriteProt = 1	phy_init	See PUB databook for details
14	Poll the PUB register MASTER.CalBusy=0	phy_init	See PUB databook for details
15	Set SWCTL.sw_done to ‘0’	phy_init	

Step	Application	SVTB Task	Notes
16	Set DFIMISC.dfi_init_start to ’1’	phy_init	
17	Set SWCTL.sw_done to ‘1’	phy_init	Require polling SWSTAT.sw_done_ackafter setting SWCTL.sw_done to ‘1’
18	Poll DFISTAT.dfi_init_complete=1	phy_init	
19	Set SWCTL.sw_done to ‘0’	phy_init	
20	Set DFIMISC.dfi_init_start to ‘0’	phy_init	
21	The following registers may need to be updated after training has completed: ■ RANKTMG0.diff_rank_wr_gap ■ RANKTMG0.diff_rank_rd_gap ■ DRAMSET1TMG2.rd2wr ■ DRAMSET1TMG2.wr2rd	phy_init	Refer to relevant PHY documentation
22	Set DFIMISC.dfi_init_complete_en to ‘1’	phy_init	
23	Set PWRCTL.selfref_sw to ‘0’	phy_init	
24	Set SWCTL.sw_done to ‘1’	phy_init	Require polling SWSTAT.sw_done_ackafter setting SWCTL.sw_done to ‘1’
25	Wait for DWC_ddrctl to move to normal operating mode by monitoring STAT.operating_mode signal	reset_dut	
26	Set back registers in step 3 to the original values if desired		

Note 1: When running training with the PHY. The following controller registers must be programmed to these at this stage:

`INITTMG0.skip_dram_init=2’b11`

`PWRCTL.selfref_sw = 1’b1`

Programming them as follows is only allowed for simulation purposes when skipping training

`INITTMG0.skip_dram_init=0` (that is, SDRAM INIT through the controller)

`PWRCTL.selfref_sw = 0`

2.2 Registers

Registers are accessed through the APB interface. The full list of registers is provided in the DesignWare Cores LPDDR5/4/4X Memory Controller Programming Guide.

The following sections outline under what circumstances the APB registers can be written. Most registers are intended to be initialized while the DDRCTL controller is in reset (`core_ddrc_rstn = 0`), and should not need to be changed afterwards. The exceptions to this rule are listed in the following sections. `core_ddrc_core_clk` must be brought up and running before DDRCTL controller is brought out of reset (`core_ddrc_rstn` is de-asserted).

The memory map is a hierarchical structure consisting of different blocks. Certain registers may be instantiated within several modules. The registers are uniquely identified by the block and register name. For example the register STAT exists within both blocks `UMCTL2_REGS` and `UMCTL2_REGS_DCH1` and the individual instances are uniquely identified as `UMCTL2_REGS.STAT` and `UMCTL2_REGS_DCH1.STAT`. The information in the following sections applies to all instances of any given register.

The "Programming Mode" field in the description column of the register tables in the "Register Descriptions" chapter. specifies the type of register, whether static, dynamic, or quasi dynamic.

This section contains the following sub-sections:

2.2.1 Dynamic Registers

The dynamic registers can be written at any time during the operation of the DDRCTL. As the APB block is typically in a different clock domain from the DDRCTL controller, these signals are synchronized to the DDRCTL controller clock domain. For the remaining registers which are considered static, synchronization is not required. To know if a particular register is of the type dynamic, see the "Programming Mode" field in the description column of the register tables in the "Register Descriptions" chapter.

2.2.1.1 Refresh Related Registers

The refresh related registers are dynamic but to update them the following must be done:

- Change the refresh associated register as desired.
- After the changed register is known stable, toggle the `RFSHCTL0.refresh_update_level` signal.

The SDRAM controller notes the `refresh_update_level` signal change and updates all refresh-related register values accordingly. This mechanism is needed to avoid sampling errors in the target clock domain as well as to allow the controller to provide special handling (such as issuing an additional refresh and resetting the refresh timer if needed) when a refresh-related timing register has changed.

Refresh related registers that can be updated are denoted as "Dynamic - Refresh Related" in the "Programming Mode" field in the "Register Descriptions" chapter.

Most of the refresh related registers are dynamic. However, in DDR5, in addition to the initialization, it is possible during self-refresh mode and MPSM as well. For details, see section "Constraints on refresh_timerX_start_value_x32" section in the DesignWare Cores LPDDR5/4/4X Memory Controller Databook. Furthermore, `RFSHCTL3.refresh_mode` can only be programmed during initialization or when the controller is in self-refresh mode. At initialization, you are required to set the `RFSHCTL0.fgr_mode` to match the refresh mode field of MR3, written to SDRAM through `INITMR0.emr3`. When updating this

register in self-refresh mode, the corresponding MR3 command is sent automatically after SRX. In this case, you must modify `INITMR0.emr3` as well, because the value written to the SDRAM through MR3 command is taken from `INITMR0.emr3`, not `RFSHCTL0.fgr_mode`.

2.2.2 Quasi Dynamic Registers

In addition to the dynamic registers, the following categories of registers can be written after reset:

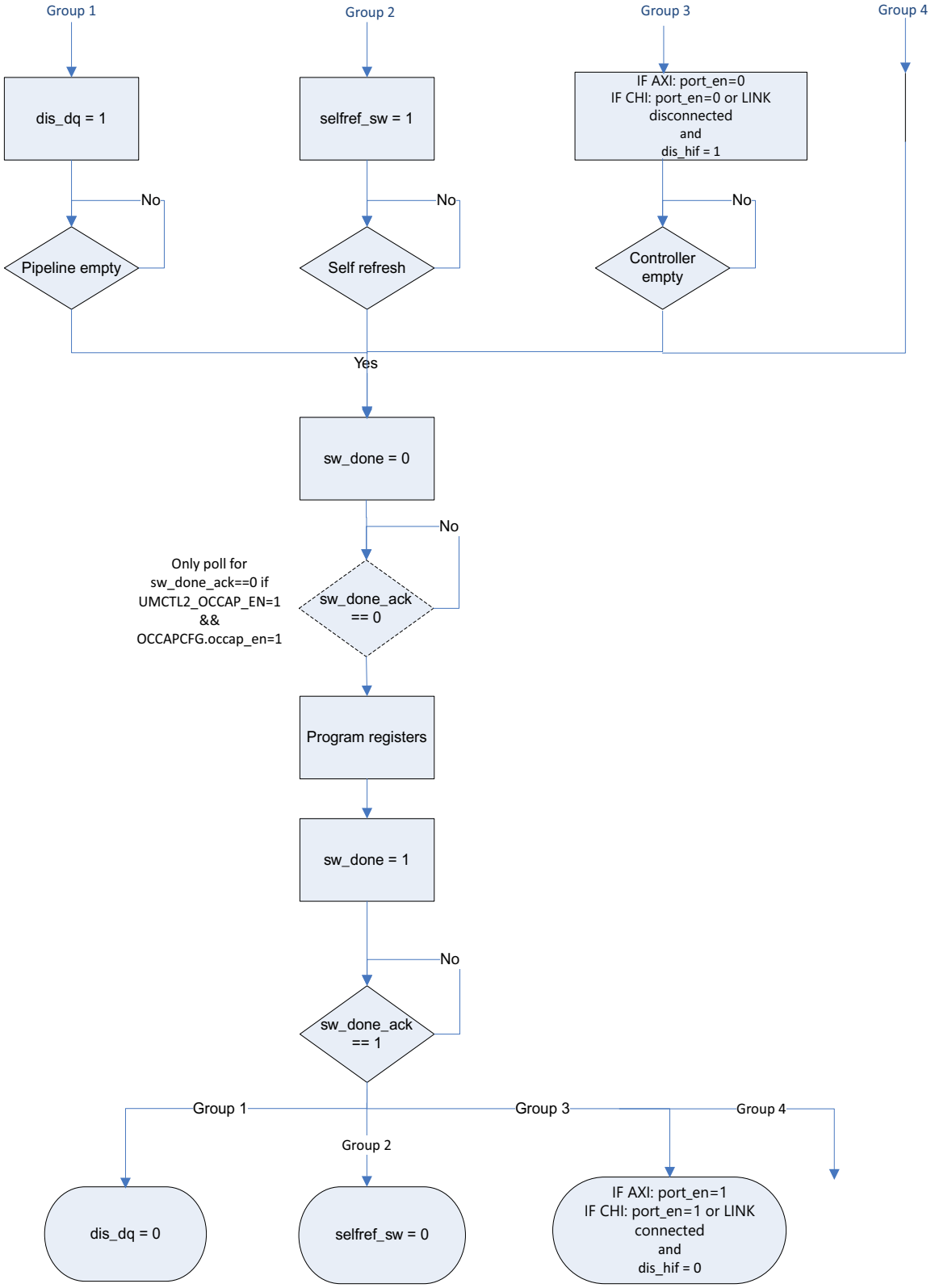
- "Group 1: Registers that can be Written when No Read/Write Traffic is Present at the DFI"
- "Group 2: Registers that can be Written in Self-refresh Mode"
- "Group 3: Registers that can be Written When Controller is Empty"
- "Group 4: Registers that can be Written Depending on MSTR2.target_frequency"

Each category requires specific conditions for the registers to be programmed. Once the programming conditions are met, `SWCTL.sw_done` register must be programmed to 1'b0 to enable the software programming. If `UMCTL2_OCCAP_EN=1 && OCCAPCFG.occap_en=1`, the `SWSTAT.sw_done_ack` must be read as 1'b0.

Once the programming is completed, `SWCTL.sw_done` must be set to 1'b1 and `SWSTAT.sw_done_ack` must be read as 1'b1. This is to ensure that quasi dynamic registers are propagated correctly to the destination clocks.

The protocol flowchart is shown in Figure 2-2.

Figure 2-2 Software Handshake Protocol



The "Programming Mode" field in the description column of the register tables in the “DWC_ddrctl Registers” chapter of the databook specifies the quasi dynamic type registers along with the group.

Traffic has to be enabled again depending on the register category as described in the following sections.

2.2.2.1 Group 1: Registers that can be Written when No Read/Write Traffic is Present at the DFI

By setting `OPCTRL1.dis_dq` register and polling `OPCTRLCAM.wr_data_pipeline_empty` and `OPCTRLCAM.rd_data_pipeline_empty`[1], it is possible to prevent any read or write traffic from being sent on the DFI. If software intervention is enabled by `CRCPARCTL1.alert_wait_for_sw`, also monitor `CRCPARSTAT.dfi_alert_err_int` and `CRCPARSTAT.dfi_alert_err_fatl_int` during the polling. If one or more of them are asserted before polling is done, retry procedure must be completed prior to the subsequent steps. In this mode, it is safe to write to the group 1 registers.

Re-enable the traffic by writing `OPCTRL1.dis_dq to 1’b0`.

2.2.2.2 Group 2: Registers that can be Written in Self-refresh Mode

When the DDRCTL has entered self-refresh mode through software (`PWRCTL.selfref_sw`), the DFI bus is idle until software exits self-refresh.

- For self-refresh, ensure that self-refresh is not caused by “Automatic Self-refresh only” by checking the `STAT.operating_mode=3’b011` and `STAT.selfref_type=2’b10`.

In this section, references to self-refresh mean SR-Powerdown.

In this mode, it is safe to write the group 2 registers.

Re-enable the traffic by writing `PWRCTL.selfref_sw` to 1’b0 and polling `STAT.operating_mode`.

2.2.2.3 Group 3: Registers that can be Written When Controller is Empty

For multi-port AXI configurations, `PCTRL.port_en` is used to enable or disable the input traffic per port.

The Controller idleness can be polled first from `PSTAT` register (`wr_port_busy_n` and `rd_port_busy_n` bit fields) and should read as `PSTAT==32’b0` (not busy). Write 0 to `SBRCTL.scrub_en` to disable SBR (required only if SBR instantiated). Poll `SBRSTAT.scrub_busy=0`, indicates that there are no outstanding SBR read commands (required only if SBR instantiated). `OPCTRL1.dis_hif` is used to enable or disable the input traffic to the controller. Then, DDRC CAM/pipeline empty status must be polled `((OPCTRLCAM.dbg_wr_q_empty== 1’b1) && (OPCTRLCAM.dbg_rd_q_empty== 1’b1) && (OPCTRLCAM.wr_data_pipeline_empty== 1’b1) && (OPCTRLCAM.rd_data_pipeline_empty== 1’b1))`[2]. If software intervention is enabled by `CRCPARCTL1.alert_wait_for_sw`, monitor `CRCPARSTAT.dfi_alert_err_int` and `CRCPARSTAT.dfi_alert_err_fatl_int` during the polling. If

1. To make sure the correct value is propagated, registers OPCTRLCAM.wr_data_pipeline_empty and OPCTRLCAM.rd_data_pipeline_empty must be polled at least twice after OPCTRL1.dis_dq is set to ‘1’.
2. To make sure the correct value is propagated, registers OPCTRLCAM.wr_data_pipeline_empty and OPCTRLCAM.rd_data_pipeline_empty must be polled at least twice after OPCTRL1.dis_hif is set to ‘1’.

one or more of them are asserted before polling is done, retry procedure must be completed prior to the subsequent steps.

In this mode, it is safe to write the group 3 registers.

Enable the traffic by writing 1'b1 to `PCTRL.port_en` or to `OPCTRL1.dis_hif` if using HIF only. Enable SBR if desired by writing 1'b1 to `SBRCTL.scrub_en` (only required if SBR instantiated).

2.2.2.4 Group 4: Registers that can be Written Depending on MSTR2.target_frequency

When `UMCTL2_FREQUENCY_NUM` is larger than 1, it is safe to write the registers in the blocks `REGB_FREQf_CHc`:

- the value of f is between 0 and (`UMCTL2_FREQUENCY_NUM -1`) excluding the register block related to the current target frequency
- the value of c is between 0 and the number of channel minus 1

For example in dual channel configurations, if `UMCTL2_FREQUENCY_NUM` is set to 3 and `MSTR2.target_frequency` is equal to '1', it is safe to write registers in `REGB_FREQ0_CH0`, `REGB_FREQ0_CH1`, `REGB_FREQ2_CH0` and `REGB_FREQ2_CH1`.

2.3 Modes of Operation

This section describes following topics:

2.3.1 SDRAM Selection

Table 2-2 SDRAM Selection

MSTR0.lpddr4	MSTR0.lpddr5	Selected DRAM
1	0	LPDDR4
0	1	LPDDR5
All other values are not supported.		

`MSTR0.lpddr4` – Present only in designs that support LPDDR4

`MSTR0.lpddr5` – Present only in designs that support LPDDR5

2.3.2 Low-Power Modes

The operating mode register bits can be polled to determine the current mode of operation of the DWC_ddrctl.

Table 2-3 shows the modes of operation.

Table 2-3 Modes of Operation

Operating Mode Register Bits	Mode of Operation of the DWC_ddrctl
000	Uninitialized. The DWC_ddrctl may be in reset, or it may be out of reset, but SDRAM initialization sequence has not yet completed.
001	Normal operating mode. The DWC_ddrctl is ready to accept read and write requests and the DWC_ddrctl may issue reads and writes to SDRAM.
010	SDRAM is in power-down mode.
011	SDRAM is in self-refresh or self-refresh power down mode.

2.4 Software Sequences

This section consists of the following subsections:

2.4.1 Changing Clock Frequencies

This section consists of the following subsections:

Note When switching from higher frequency to lower frequency, the controller may violate the JEDEC requirement that no more than 16 refreshes must be issued within 2*tREFI. These extra refreshes are not expected to cause a problem in the SDRAM.

2.4.1.1 Changing Clock Frequencies Without Frequency Set Point

The operations for frequency change flow involve the following steps:

1. During initialization, program the timing register-set REGB_FREQ(1/2/3)_CH, whichever you prefer, with timing settings required for the alternative clock frequency.
2. Set `PCTRL.port_en` to '0' to block the AXI ports from taking transactions. Poll `PSTAT.rd_port_busy_n=0` and `PSTAT.wr_port_busy_n=0`. Wait until all AXI ports are idle.
3. If Scrubber is instantiated, set `SBRCTL.scrub_en` to '0', to disable the scrubber and poll `SBRSTAT.scrub_busy=0`. This indicates that, there are no outstanding SBR read commands.
4. Set `OPCTRL1.dis_hif` to '1', so that no new commands are accepted by the controller and poll `OPCTRLCAM.dbg_wr_q_empty=1`, `OPCTRLCAM.wr_data_pipeline_empty=1`, `OPCTRLCAM.dbg_rd_q_empty=1`, `OPCTRLCAM.rd_data_pipeline_empty=1`.
5. If `DERATECTL0.derate_eanble` is '1', then set `DERATECTL0.derate_mr4_pause_fc` to '1' to pause automatic MRR to MR4.
6. If `DFIUPD0.dis_auto_ctrlupd = 0`, then set `DFIUPD0.dis_auto_ctrlupd` to '1' to disable periodic DFI update request temporarily.
7. Set `OPCTRLCMD.ctrlupd` to '1' and poll `OPCTRLSTAT.ctrlupd_busy = '0'` to issue DFI update request manually.
8. If DFI PHY Master interface is active in controller (`DFIPHYMSTR.dfi_phymstr_en`[1] `= 1`) then disable it by programming `DFIPHYMSTR.dfi_phymstr_en` to '0'.

1. The register DFIPHYMSTR.phymstr_en is a static register and therefore is normally set while the controller is in reset. However, it may be set outside of reset as part of this sequence.

9. Disable DQS Oscillator if it is enabled, by setting `DQSOSCCTL0.dqsosc_enable` to '0'. Poll `DQSOSCSTAT0.dqsosc_state = 0`.
10. Disable low-power activities by programming `PWRCTL.selfref_en`, `PWRCTL.powerdown_en` and `HWLPCTL.hw_lp_en` to '0'.
11. Poll `STAT.operating_mode=1`, which indicates the controller is in normal mode.
12. If `DFILPCFG0.dfi_lp_en_sr`[1] is '1', set `DFILPCFG0.dfi_lp_en_sr` to '0' and poll `DFISTAT.dfi_lp_ctrl_ack_stat=0`.

> **Note** `DFILPCFG0.dfi_lp_en_sr` can be disabled only in normal mode or power down with `PWRCTL.selfref_en=0` to avoid both entering self refresh power down and internal DFI low power happen at the same time.

13. If `DFILPCFG0.dfi_lp_data_req_en` is '1', set `DFILPCFG0.dfi_lp_data_req_en` to '0' and poll `DFISTAT.dfi_lp_data_ack_stat = 0`.
14. Set `PWRCTL.en_dfi_dram_clk_disable` to '0'.
15. Enter self-refresh mode by setting `PWRCTL.selfref_sw` to '1'. Poll `operating_mode = 3`, which indicates controller entered into self refresh.
16. Ensure that self-refresh is due to software by checking that `STAT.selfref_type[1:0] = 2'b10`.
17. Set `OPCTRL1.dis_dq` to '1' so that no CAM commands are de-queued. Poll `OPCTRLCAM.wr_data_pipeline_empty=1`, `OPCTRLCAM.rd_data_pipeline_empty=1`.
18. Set `DFIMISC.dfi_init_complete_en` to '0' to ensure that the controller initialization state machine is not reset if the PHY needs to perform some initialization after the frequency change.
19. If DFI PHY update interface is active in the controller (`DFIUPD0.dfi_phyupd_en = 1`), disable it by programming `DFIUPD0.dfi_phyupd_en` to '0'.
20. Program `DFIMISC.dfi_frequency` to the `target_frequency` value. (Refer to PHY databook for more details.)
21. Set `DFIMISC.dfi_init_start` to '1'. PHY performs internal sequences to cleanly stop pipelines and prepare for clock frequency change.
22. The PHY de-asserts `dfi_init_complete`. Poll `DFISTAT.dfi_init_complete=0`.
23. Change the clock frequency to the controller ensuring there are no glitches.
24. Program `MSTR2.target_frequency` to the `target_frequency` value, to select `REGB_FREQ*` registers.
25. Toggle `RFSHCTL0.refresh_update_level` to allow the new refresh-related register values to propagate to the refresh logic.
26. Set `DFIMISC.dfi_init_start` to '0'. The PHY performs internal sequences to relock PLLs and calibrate ZQ/Delay-Lines.
27. If the DFI PHY update interface was active prior to frequency change, then enable PHY update Interface by setting `DFIUPD0.dfi_phyupd_en` to '1'.
28. The PHY asserts `dfi_init_complete`. Poll `DFISTAT.dfi_init_complete=1`.

1. The register DFILPCFG0.dfi_lp_en_sr is a static register and therefore is normally set while the controller is in reset. However, it may be set outside of reset as part of this sequence.

29. Set DFIMISC.dfi_init_complete_en to '1', to indicate to the controller that the PHY has finished frequency change.
30. Exit self-refresh by setting PWRCTL.selfref_sw to '0'. And wait until STAT.operating_mode != 2'b11, indicating controller exited self-refresh state.
31. Restore the values of DFILPCF0.dfi_lp_en_sr and DFILPCFG0.dfi_lp_data_req_en after to self-refresh exit.
32. Update Mode Registers of the DRAM (through MRCTRL0.mr_* / MRCTRL1.mr_*) if necessary due to timing values change because of frequency change.
33. Set OPCTRL1.dis_dq back to '0', to allow read and write traffic to be sent to SDRAM.
34. Program the controller registers back to their default value, which were disabled prior to frequency change.
35. Enable HIF commands by setting OPCTRL1.dis_hif to '0'.
36. Reset DERATECTL0.derate_mr4_pause_fc to '0' if DERATECTL0.derate_mr4_pause_fc has been set to '1' prior to frequency change.
37. Reset DFIUPD0.dis_auto_ctrlupd to '0' if DFIUPD0.dis_auto_ctrlupd has been set to '1' prior to frequency change.
38. If the DFI PHY Master interface was active prior to frequency change, then enable PHY Master interface by setting DFIPHYMSTR.dfi_phymstr_en to '1'.
39. Enable DQS Oscillator if it was disabled prior to frequency change, by setting DQSOSCCTL0.dqsosc_enable to '1'.
40. Enable PWRCTL.selfref_en, PWRCTL.powerdown_en, HWLPCTL.hw_lp_en if it was disabled prior to frequency change.
41. Set PWRCTL.en_dfi_dram_clk_disable to '1' if it was programmed to '0' prior to frequency change.
42. Enable AXI ports by programming PCTRL.port_en to '1' and scrubber by programming SBRCTL.scrub_en to '1', if AXI ports and scrubber were disabled prior to frequency change.

2.4.1.2 Changing Clock Frequencies With Frequency Set Point

The operations for frequency change flow involve the following steps:

1. During initialization, program the timing register-set REGB_FREQ(1/2/3)_CH, whichever you prefer, with the timing settings required for the alternative clock frequency.
2. Set PCTRL.port_en to '0' to block the AXI ports from taking the transactions. Poll PSTAT.rd_port_busy_n=0 and PSTAT.wr_port_busy_n=0. Wait until all AXI ports are idle.
3. If Scrubber is instantiated, set SBRCTL.scrub_en to '0', to disable the scrubber and poll SBRSTAT.scrub_busy=0. This indicates that there are no outstanding SBR read commands.
4. Set OPCTRL1.dis_hif to '1', so that no new commands are accepted by the controller and poll OPCTRLCAM.dbg_wr_q_empty=1, OPCTRLCAM.wr_data_pipeline_empty=1, OPCTRLCAM.dbg_rd_q_empty=1, OPCTRLCAM.rd_data_pipeline_empty=1.
5. If DERATECTL0.derate_eanble is '1', then set DERATECTL0.derate_mr4_pause_fc to '1' to pause automatic MRR to MR4.
6. If DFIUPD0.dis_auto_ctrlupd = 0, then set DFIUPD0.dis_auto_ctrlupd to '1' to disable periodic DFI update request temporarily.

7. Set `OPCTRLCMD.ctrlupd` to '1' and poll `OPCTRLSTAT.ctrlupd_busy = 0` to issue DFI update request manually.
8. If DFI PHY Master interface is active in the controller (`DFIPHYMSTR.dfi_phymstr_en = 1`), then disable it by programming `DFIPHYMSTR.dfi_phymstr_en` to '0'.
9. Disable DQS Oscillator if it is enabled, by setting DQSOSCCTL0.dqsosc_enable to '0'. Poll DQSOSC-STAT0.dqsosc_state= '0'.
10. Disable automatic ZQ Calibration if it is enabled, by setting `ZQCTL0.dis_auto_zq` to '1'.
11. Disable low-power activities by programming `PWRCTL.selfref_en`, `PWRCTL.powerdown_en` and `HWLPCTL.hw_lp_en` to '0'.
12. Poll `STAT.operating_mode=1`, which indicates the controller is in normal mode.
13. If `DFILPCFG0.dfi_lp_en_sr` is '1', set `DFILPCFG0.dfi_lp_en_sr` to '0' and poll `DFISTAT.dfi_lp_ctrl_ack_stat = 0`.

Note `DFILPCFG0.dfi_lp_en_sr` can be disabled only in normal mode or power down with `PWRCTL.selfref_en = 0` to avoid both entering self refresh power down and internal DFI low power happen at the same time.

14. If `DFILPCFG0.dfi_lp_data_req_en` is '1', set `DFILPCFG0.dfi_lp_data_req_en` to '0' and poll `DFISTAT.dfi_lp_data_ack_stat = 0`.
15. Set `PWRCTL.en_dfi_dram_clk_disable` to '0'.
16. Set `PWRCTL.stay_in_selfref` to '1' and then set `PWRCTL.selfref_sw` to '1'.
17. Poll `STAT.selfref_state != 2'b00`, indicating the controller entered into self-refresh state.
18. Read `STAT.selfref_state` and `STAT.selfref_type`. If `STAT.selfref_state = 2'b01` and `STAT.selfref_type != 2'b01` ('Self Refresh 1' state caused solely by the software, not by DFI PHY Master hardware state machine), then jump to the next step, else program `PWRCTL.selfref_sw` to '0' and program `PWRCTL.stay_in_selfref` to '0', and return to step 16.
19. Set `OPCTRL1.dis_dq` to '1' so that no CAM commands are de-queued. Poll `OPCTRLCAM.wr_data_pipeline_empty = 1`, `OPCTRLCAM.rd_data_pipeline_empty = 1`.
20. In LPDDR4, set `PWRCTL.stay_in_selfref` to '0' to enter 'Self-refresh power-down' and poll `STAT.selfref_state = 2'b10`. This indicates the controller entered into self-refresh power-down state.
21. Set `DFIMISC.dfi_init_complete_en` to '0' to ensure that the controller initialization state machine is not reset if the PHY needs to perform some initialization after the frequency change.
22. If DFI PHY update interface is active in the controller (`DFIUPD0.dfi_phyupd_en = 1`), disable it by programming `DFIUPD0.dfi_phyupd_en` to '0'.
23. Program `DFIMISC.dfi_frequency` to the `target_frequency` value (refer to PHY databook for more details).
24. Toggle `DFIMISC.dfi_freq_fsp[0]` to select a different DRAM FSP for the target PState.
25. Set `DFIMISC.dfi_init_start` to '1'. PHY performs internal sequences to cleanly stop pipelines and prepare for clock frequency change.
26. The PHY de-asserts `dfi_init_complete`. `Poll DFISTAT.dfi_init_complete = 0`.
27. Change the clock frequency to the controller ensuring there are no glitches.

28. Program `MSTR2.target_frequency` to the `target_frequency` value, to select `REGB_FREQ*` registers.
29. Toggle `RFSHCTL0.refresh_update_level` to allow the new refresh-related register values to propagate to the refresh logic.
30. Set `DFIMISC.dfi_init_start` to '0'. The PHY performs internal sequences to relock PLLs and calibrate ZQ/Delay-Lines.
31. If the DFI PHY update interface was active prior to frequency change, then enable PHY update Interface by setting `DFIUPD0.dfi_phyupd_en` to '1'.
32. The PHY asserts `dfi_init_complete`. Poll `DFISTAT.dfi_init_complete = 1`.
33. Set `DFIMISC.dfi_init_complete_en` to '1', to indicate to the controller that PHY finished frequency change.
34. Set `ZQCTL2.dis_srx_zqcl` to '1', if `ZQCTL2.dis_srx_zqcl = 0`.
35. In LPDDR4, set `PWRCTL.stay_in_selfref` to '1', and then set `PWRCTL.selfref_sw` to '0'. Poll `STAT.selfref_state = 2'b11`. This indicates the controller entered into 'Self refresh 2' state.
36. If DVFSQ[1] is not active (MR19.OP[3:2] = 0), issue `zq_calib_short` command by programming `OPCTRLCMD.zq_calib_short` to '1' and poll `OPCTRLSTAT.zq_calib_short_busy = 0`.
37. If DVFSQ[2] is not active (MR19.OP[3:2] = 0), set `ZQCTL2.dis_srx_zqcl` to '0', if it was set to '1' prior to frequency change.
38. In LPDDR5, set `PWRCTL.selfref_sw` to '0'.
39. Set `PWRCTL.stay_in_selfref` to '0'. Poll `STAT.selfref_state = 0`. This indicates that the controller is not in self-refresh state.
40. After self-refresh exit, restore the values of registers `DFILPCF0.dfi_lp_en_sr` and `DFILPCFG0.dfi_lp_data_req_en` to its original value.
41. Set `OPCTRL1.dis_dq` back to '0', to allow read and write traffic to be sent to SDRAM.
42. Program the controller registers back to their default values, which were disabled prior to frequency change.
43. Enable HIF interface by setting `OPCTRL1.dis_hif` to '0'.
44. Reset `DERATECTL0.derate_mr4_pause_fc` to '0' if `DERATECTL0.derate_mr4_pause_fc` has been set to '1' prior to frequency change.
45. Reset `DFIUPD0.dis_auto_ctrlupd` to '0' if `DFIUPD0.dis_auto_ctrlupd` has been set to '1' prior to frequency change.
46. If the DFI PHY Master interface was active prior to frequency change, then enable PHY Master interface by setting `DFIPHYMSTR.dfi_phymstr_en` to '1'.
47. Enable DQS Oscillator if it was disabled prior to frequency change, by setting `DQSOSCCTL0.dqsosc_enable` to '1'.
48. If DVFSQ[3] is not active (MR19.OP[3:2] = 0), enable automatic ZQ Calibration if it was disabled prior to frequency change, by setting `ZQCTL0.dis_auto_zq` to '0'.

1. DVFSQ is applicable only for LPDDR5. Execute this step for LPDDR4 as DVFSQ is not applicable and LPDDR5 if DVFSQ is not active.
2. DVFSQ is applicable only for LPDDR5. Execute this step for LPDDR4 as DVFSQ is not applicable and LPDDR5 if DVFSQ is not active.
3. DVFSQ is applicable only for LPDDR5. Execute this step for LPDDR4 as DVFSQ is not applicable and LPDDR5 if DVFSQ is not active.

49. Enable PWRCTL.selfref_en, PWRCTL.powerdown_en, HWLPCTL.hw_lp_en if it was disabled prior to frequency change.
50. Set PWRCTL.en_dfi_dram_clk_disable to '1' if it was programmed to '0' prior to frequency change.
51. Enable AXI ports by programming PCTRL.port_en to '1', and scrubber by programming SBRCTL.scrub_en to '1', if AXI ports and scrubber were disabled prior to frequency change.

Note

- Ensure to satisfy the timing parameters tCKFSPE, tFC, tCKFSPX.
- When using dfi_freq_fsp pin to switch the FSP-OP, Synopsys DWC LPDDR5/4/4X PHY will leave the SDRAM at FSP-1 at the end of initialization (training).

  So, the following settings will be present.

  During initialization, MSTR2.target_frequency (0) = DFIMISC.dfi_frequency (0) = DFIMISC.dfi_freq_fsp (1) = SDRAM.FSP_OP (1). And SDRAM MR values will be programmed as per the controller's FREQ-0 register values.

  During frequency change, MSTR2.target_frequency (1) = DFIMISC.dfi_frequency (1) = DFIMISC.dfi_freq_fsp (0) = SDRAM.FSP_OP (0). And SDRAM MR values will be programmed as per the controller's FREQ-1 register values.
- After setting dfi0_init_start = 1, DWC LPDDR5/4/4X PHY sets VRCG = 1 in MR13 (LPDDR4) or MR16 (LPDDR5) before dfi0_init_complete becomes '0'.
- After setting dfi0_init_start = 0, DWC LPDDR5/4/4X PHY sets VRCG = 0 in MR13 (LPDDR4) or MR16 (LPDDR5) before dfi0_init_complete becomes '1'.

2.4.1.3 Changing Clock Frequencies With Enabling/Disabling DVFSC

Note

The JESD209-5A specification does not explicitly mention that DVFSC mode change is allowed only in Idle state. However, note that some SDRAM does not allow to change DVFSC mode during Self-Refresh/Self-Refresh Power-Down because power rail change during Self-Refresh has a negative impact on Self-Refresh control logic in SDRAM. In this case, it is not recommended to use this software sequence.

During initialization, set LPDDR5 MR19 mode register to the desired value to enable or disable DVFSC. Then perform the steps described in section "Changing Clock Frequencies With Frequency Set Point" on page 676. Synopsys DWC LPDDR5/4/4X PHY will program the SDRAM mode registers during initialization and frequency change.

For example, changing the data_rate from 6400Mbps (FREQ_0: DVFSC disable) to 1067Mbps (FREQ_1: DVFSC enable).

FREQ_0: MR19.OP[1:0] = 0 (to use VDD2H: 1.05V rail - {DVFSC disable}).

FREQ_1: MR19.OP[1:0] = 1 (to use VDD2L: 0.9V rail - {DVFSC enable}).

2.4.1.4 Changing Clock Frequencies With Enabling/Disabling DVFSQ

The operations for frequency change flow involve the following steps:

DVFSQ High-to-Low Transition

1. During initialization, set LPDDR5 MR19 mode register to the desired value to enable or disable DVFSQ. In this case, set MR19.OP[3:2] = 1 (to use VDDQ = 0.3V - {DVFSQ LOW}). Synopsys DWC LPDDR5/4/4X PHY will program the SDRAM mode registers during initialization and frequency change.
2. Execute the frequency change steps till step 34 described in section "Changing Clock Frequencies With Frequency Set Point" on page 676 (step 34: Set `ZQCTL2.dis_srx_zqcl` to '1', if `ZQCTL2.dis_srx_zqcl = 0`).
3. Poll `ZQSTAT.zq_reset_busy = 0`. The SoC can initiate a ZQ Reset operation only if `ZQSTAT.zq_reset_busy` is low.
4. Set `ZQCTL1.zq_reset` to '1' and poll `ZQSTAT.zq_reset_busy = 0`. This sets MR28.OP[0] = 1(ZQ RESET) at SDRAM, to set default ZQ strength to meet VDDQ = 0.3V operation.
5. Set MR28.OP[1](ZQ Stop) to '1', through Mode Register Read/Write signals (`MRCTRL0.mr_*/MRCTRL1.mr_*`), to disable background calibration.
6. Wait for tZQSTOP time.
7. Enter DVFSQ (reduce VddQ below 0.5v nominal).
8. Continue operation with reduced VddQ.
9. Continue frequency change steps from step 38 described in section "Changing Clock Frequencies With Frequency Set Point" on page 676 (step 38: In LPDDR5, set `PWRCTL.selfref_sw` to '0').

Note `VRCG = 0` in MR16 is set by DWC LPDDR5/4/4X PHY as part of the frequency change procedure while `dfi_init_start = 0` and `dfi_init_complete = 0`. This is aligned with section 7.7.1.2.1 of JESD209-5A.

DVFSQ Low-to-High Transition Without VRCG During VddQ Ramp

1. During initialization, set LPDDR5 MR19 mode register to the desired value to enable or disable DVFSQ. In this case, set MR19.OP[3:2] = 0 (to use VDDQ = 0.5V - {DVFSQ HIGH}). Synopsys DWC LPDDR5/4/4X PHY will program the SDRAM mode registers during initialization and frequency change.
2. Execute the frequency change steps till step 19 described in section "Changing Clock Frequencies With Frequency Set Point" on page 676 (step 19: Set `OPCTRL1.dis_dq` to '1' so that no CAM commands are de-queued. Poll `OPCTRLCAM.wr_data_pipeline_empty = 1`, `OPCTRLCAM.rd_data_pipeline_empty = 1`).
3. SDRAM will be operating at low speed with VddQ<0.5v nominal when DVFSQ is LOW.
4. Ramp VddQ up to 0.5v nominal.
5. Set MR28.OP[1](ZQ Stop) to '0', through Mode Register Read/Write signals (`MRCTRL0.mr_*/MRCTRL1.mr_*`), to enable background calibration.
6. Wait tZQCALx.
7. Continue frequency change steps from step 21 described in section "Changing Clock Frequencies With Frequency Set Point" on page 676 (step 21: Set `DFIMISC.dfi_init_complete_en` to '0' to ensure that the controller initialization state machine is not reset if the PHY needs to perform some initialization after the frequency change).

DVFSQ Low-to-High Transition with VRCG

1. During initialization, set LPDDR5 MR19 mode register to the desired value to enable or disable DVFSQ. In this case, set MR19.OP[3:2] = 0 (to use VDDQ = 0.5V - {DVFSQ HIGH}). Synopsys DWC LPDDR5/4/4X PHY will program the SDRAM mode registers during initialization and frequency change.
2. Execute the frequency change steps till step 19 described in section "Changing Clock Frequencies With Frequency Set Point" on page 676 (step 19: Set `OPCTRL1.dis_dq` to '1' so that no CAM commands are de-queued. Poll `OPCTRLCAM.wr_data_pipeline_empty = 1`, `OPCTRLCAM.rd_data_pipeline_empty = 1`).
3. SDRAM will be operating at low speed with VddQ<0.5v nominal when DVFSQ is LOW.
4. Enable VRCG (MR16.OP[6] = 1), through Mode Register Read/Write signals (`MRCTRL0.mr_*/MRCTRL1.mr_*`).
5. Wait `tVRCG_enable` (stall traffic).
6. Ramp VddQ up to 0.5v nominal.
7. Set MR28.OP[1](ZQ Stop) to '0', through Mode Register Read/Write signals (`MRCTRL0.mr_*/MRCTRL1.mr_*`), to enable background calibration.
8. Wait tZQCALx.
9. Continue frequency change steps from step 21 described in section "Changing Clock Frequencies With Frequency Set Point" on page 676 (step 21: Set `DFIMISC.dfi_init_complete_en` to '0' to ensure that the controller initialization state machine is not reset if the PHY needs to perform some initialization after the frequency change).

Note For more details on ZQRESET and ZQSTOP, refer to section "ZQ Calibration" in LPDDR5/4/4X Memory Controller Databook.

2.4.2 Software Sequence for Removal of Clocks

Software can be used to keep the SDRAM in self-refresh. The AXI and DDRC clocks can be removed when in self-refresh by following the sequence described in Table 2-4.

Note Dynamic and quasi dynamic registers can not be programmed if any of the clocks has been removed. Clocks must be turned back on before starting the programming sequence. Also the clock gating logic must ensure there are no glitches on the clocks when there are removed/enabled.

In the following software sequences, ignore the registers that are not available for your configuration. You can generate a report of registers for your configuration after you configure the core in the coreConsultant GUI. See "Creating Optional Reports and Views".

Table 2-4 Software Clock Removal Sequence

Step	Description	Comment
1	Write '0' to PCTRL.port_en	Blocks AXI port from taking anymore transactions.

Step	Description	Comment
2	Poll PSTAT.rd_port_busy_n = 0 Poll PSTAT.wr_port_busy_n = 0	Waits until all AXI ports are idle.
3	Write ‘0’ to SBRCTL.scrub_en	Disables SBR, only required if SBR is instantiated.
4	Poll SBRSTAT.scrub_busy = 0	Indicates that there are no outstanding SBR read commands, only required if SBR instantiated.
5	Write ‘1’ to PWRCTL.selfref_sw	Causes the system to move to Self Refresh state.
6	Poll STAT.selfref_type= 2’b10 Poll STAT.selfref_state = 3'b010	Waits until Self Refresh state is entered. Waits until Self Refresh Power Down state is entered.
7	Remove AXI clock	
8	Remove DDRC controller clock	

The clocks must be re-enabled by following the sequence described in Table 2-5.

Table 2-5 Re-enabling the clocks

Step	Description	Comment
1	Enable AXI clock	
2[a]	Enable DDRC controller clock	
3	Write ‘0’ to PWRCTL.selfref_sw	Cause system to exit from self-refresh state.
4	Poll STAT.selfref_type = 2’b00	Wait until self-refresh state is exited.
5	Write ‘1’ to PCTRL.port_en	AXI ports are no longer blocked from taking transactions.
6	Write ‘1’ to SBRCTL.scrub_en	Enable SBR if desired, only required if SBR instantiated.

a. After this step, it is recommended for DDR4 to re-enable generation of PHY initiated Update requests (dfi_phyupd_req). If using a Synopsys PHY, this can be done through the PUB’s DSGCR.PUREN.

2.4.3 Power Removal Flow

Table 2-6 Power Removal

Step	Description	Comment
1	Write ‘0’ to PCTRL_n.port_en	Blocks AXI port from taking anymore transactions.
2	Poll PSTAT.rd_port_busy_n = 0 Poll PSTAT.wr_port_busy_n = 0	Waits unit all AXI ports are idle.
3	Write ‘0’ to SBRCTL.scrub_en	Disables SBR, only required if SBR instantiated.
4	Poll SBRSTAT.scrub_busy = 0	Indicates that there are no outstanding SBR read commands, only required if SBR instantiated.

Step	Description	Comment
5	Set ZQ Stop(MR28 OP[1]) to ’1’. Send MRW command using MRCTRL0 and MRCTRL1. (LPDDR5 only)	For LPDDR5, ZQ Stop needs to be set to ’1’, if VDDQ needs to be turned off during self-refresh power down state.
6	Write ‘1’ to PWRCTL.selfref_sw	Causes system to move to self-refresh state.
7	Poll STAT.selfref_type= 2’b10 Poll STAT.selfref_state = 3'b010	Waits until self-refresh power down state is entered.
8	Program DFIMISC.dfi_frequency as required. Refer to PHY databook for more details.	0x1f for PHY deep-sleep/retention.
9	Set DFIMISC.dfi_init_start to ‘1’	
10	The PHY de-asserts dfi_init_complete. The controller polls DFISTAT.dfi_init_complete=0.	
11	Set DFIMISC.dfi_init_start to ‘0’.	
	The PHY asserts dfi_init_complete.The controller polls DFISTAT.dfi_init_complete=1.	
12	Place IOs in retention mode	Refer to relevant PHY databook for more information.
13	Remove power	

Table 2-7 Re-enabling the Power

Step	Description	Comment
1	Enable Power	
2	Reset controller/PHY by driving core_ddrc_rstn = 1’b0, aresetn_n = 1’b0, presetn = ’b0	
3	Remove APB reset, presetn = 1’b1, and reprogram the registers to pre-power removal values	
4	Program RFSHCTL0.dis_auto_refresh =1'b1	
5	Program INITTMG0.skip_dram_init = 2’b11	Skips the DRAM initialization routine and starts up in self-refresh mode.
6	Programs PWRCTL.selfref_sw = 1’b1	Keeps the controller in self-refresh mode.

Step	Description	Comment
7	Program DFIMISC.dfi_init_complete_en to 1’b0	PHY initialization needs to be rerun so set to ’0’ until initialization complete.
8	Remove the controller reset core_ddrc_rstn = 1’b1 aresetn_n = 1’b1	
9	Program RFSHCTL0.dis_auto_refresh =1'b0	
10	Run PHY initialization/training as required, including removing the IOs from retention mode including steps below	Refer to the relevant PHY databook for more information.
11	Program DFIMISC.dfi_frequency as required. Refer to PHY databook for more details	
12	Set DFIMISC.dfi_init_start to ‘1’	
13	The PHY asserts dfi_init_complete. The controller polls `DFISTAT.dfi_init_complete=1`.	
14	Set DFIMISC.dfi_init_start to ‘0’.	
15	Program DFIMISC.dfi_init_complete_en to 1'b1	Indicates to controller that PHY has completed re-training/initialization.
16	Program PWRCTL.selfref_sw = 1’b0	Trigger self-refresh exit.
17	Poll STAT.selfref_type = 2’b00	Wait until self-refresh state is exited.
18	Poll STAT.operating_mode for Normal Mode entry	
19	Set ZQ Stop(MR28 OP[1]) to ’0’. Send MRW command using MRCTRL0 and MRCTRL1. (LPDDR5 only)	For LPDDR5, ZQ Stop needs to be set to ’0’, if ZQ Stop is set to ’1’ before power removal.
20	Write PCTRL.port_en = 1	AXI ports are no longer blocked from taking transactions.
21	Write ‘1’ to SBRCTL.scrub_en	Enable SBR if desired, only required if SBR instantiated.

2.4.4 Special Software Self-Refresh Sequence With Controller Initiated Retraining Followed by Burst PPT2

The operations for software self-refresh along with retraining (PHY retraining and Burst PPT2) on SWSR exit involve the following steps:

1. During initialization, PPT2 related registers are assumed to be set, including `DFIUPDTMG2.ppt2_en = 1`.
2. Set `PCTRL.port_en` to '0' to block the AXI ports from taking transactions. Poll `PSTAT.rd_port_busy_n = 0` and `PSTAT.wr_port_busy_n = 0`. Wait until all AXI ports are idle.
3. If Scrubber is instantiated, set `SBRCTL.scrub_en` to '0', to disable the scrubber and poll `SBRSTAT.scrub_busy = 0`. This indicates that there are no outstanding SBR read commands.
4. Set `OPCTRL1.dis_hif` to '1', so that no new commands are accepted by the controller and poll `OPCTRLCAM.dbg_wr_q_empty = 1`, `OPCTRLCAM.wr_data_pipeline_empty = 1`, `OPCTRLCAM.dbg_rd_q_empty = 1`, `OPCTRLCAM.rd_data_pipeline_empty = 1`.
5. If `DERATECTL0.derate_eanble` is '1', then set `DERATECTL0.derate_mr4_pause_fc` to '1' to pause automatic MRR to MR4.
6. If `DFIUPD0.dis_auto_ctrlupd = 0`, then set `DFIUPD0.dis_auto_ctrlupd` to '1' to disable periodic DFI update request temporarily.
7. Set `OPCTRLCMD.ctrlupd` to '1' and poll `OPCTRLSTAT.ctrlupd_busy = 0` to issue DFI update request manually.
8. Disable low-power activities by programming `PWRCTL.selfref_en`, `PWRCTL.powerdown_en` and `HWLPCTL.hw_lp_en` to '0'.
9. Poll `STAT.operating_mode = 1`, which indicates the controller is in normal mode.
10. If `DFILPCFG0.dfi_lp_en_sr` is '1', set `DFILPCFG0.dfi_lp_en_sr` to '0' and poll `DFISTAT.dfi_lp_ctrl_ack_stat = 0`.

> **Note** `DFILPCFG0.dfi_lp_en_sr` can be disabled only in normal mode or power-down with `PWRCTL.selfref_en = 0` to avoid both entering self-refresh power-down and internal DFI low-power happen at the same time.

11. If `DFILPCFG0.dfi_lp_data_req_en` is '1', set `DFILPCFG0.dfi_lp_data_req_en` to '0' and poll `DFISTAT.dfi_lp_data_ack_stat = 0`.
12. Set `PWRCTL.en_dfi_dram_clk_disable` to '0'.
13. Enter self-refresh mode by setting `PWRCTL.selfref_sw` to '1'. Poll `operating_mode = 3`, which indicates the controller entered into self-refresh.
14. Ensure that self-refresh is due to software by checking that `STAT.selfref_type[1:0] = 2'b10`.
15. Set `OPCTRL1.dis_dq` to '1' so that no CAM commands are de-queued. Poll `OPCTRLCAM.wr_data_pipeline_empty = 1`, `OPCTRLCAM.rd_data_pipeline_empty = 1`.
16. Set `DFIMISC.dfi_init_complete_en` to '0' to ensure that the controller initialization state machine is not reset if the PHY needs to perform some initialization while retraining.
17. Program `DFIMISC.dfi_frequency` 5'h17, that is Retrain only.
18. Long term Idle.
19. Set `DFIMISC.dfi_init_start` to '1'. PHY performs internal sequences to cleanly stop pipelines.

20. The PHY de-asserts `dfi_init_complete`. Poll `DFISTAT.dfi_init_complete = 0`.
21. Set `DFIMISC.dfi_init_start` to '0'. The PHY performs internal sequences to re-lock PLLs and calibrate ZQ/Delay-Lines.
22. The PHY asserts `dfi_init_complete`. Poll `DFISTAT.dfi_init_complete = 1`.
23. Set `DFIMISC.dfi_init_complete_en` to '1', to indicate to the controller that the PHY has finished retraining.
24. Start the Burst PPT2 process. If `ZQCTL0.dis_auto_zq` is '0', set `ZQCTL0.dis_auto_zq` to '1'.
25. Program `PPT2CTRL0.ppt2_burst` to ‘1’.
26. Exit self-refresh by setting `PWRCTL.selfref_sw` to '0'. And wait until `STAT.operating_mode != 2'b11`, indicating the controller exited self-refresh state.
27. Once the `selfref_sw` is made, the Burst PPT2 process starts. Wait for Burst PPT2 to complete by polling `PPT2STAT0.ppt2_burst_busy` to ‘0’.
28. Reset `ZQCTL0.dis_auto_zq` to '0' if `ZQCTL0.dis_auto_zq` has been set to '1' prior to Burst PPT2.
29. Restore the values of `DFILPCF0.dfi_lp_en_sr` and `DFILPCFG0.dfi_lp_data_req_en` after the self-refresh exit.
30. Set `OPCTRL1.dis_dq` back to '0', to allow read and write traffic to be sent to SDRAM.
31. Program the controller registers back to their default values, which were disabled prior to retraining.
32. Enable HIF commands by setting `OPCTRL1.dis_hif` to '0'.
33. Reset `DERATECTL0.derate_mr4_pause_fc` to '0' if `DERATECTL0.derate_mr4_pause_fc` has been set to '1' prior to retraining.
34. Reset `DFIUPD0.dis_auto_ctrlupd` to '0' if `DFIUPD0.dis_auto_ctrlupd` has been set to '1' prior to retraining.
35. Enable `PWRCTL.selfref_en`, `PWRCTL.powerdown_en`, `HWLPCTL.hw_lp_en` if they were disabled prior to retraining.
36. Set `PWRCTL.en_dfi_dram_clk_disable` to '1' if it was programmed to '0' prior to retraining.
37. Enable AXI ports by programming `PCTRL.port_en` to '1' and scrubber by programming `SBRCTL.scrub_en` to '1', if AXI ports and scrubber were disabled prior to retraining.