DREX-1 v2.0

High-Performance SDRAM Controller

Revision 2.0

Sep 2011

User's Guide

SAMSUNG ELECTRONICS

SAMSUNG

Important Notice

The information in this publication has been carefully checked and is believed to be entirely accurate at the time of publication. Samsung assumes no responsibility, however, for possible errors or omissions, or for any consequences resulting from the use of the information contained herein.

Samsung reserves the right to make changes in its products or product specifications with the intent to improve function or design at any time and without notice and is not required to update this documentation to reflect such changes.

This publication does not convey to a purchaser of semiconductor devices described herein any license under the patent rights of Samsung or others.

Samsung makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Samsung assume any liability arising out of the application or use of any product or circuit and specifically disclaims any and all liability, including without limitation any consequential or incidental damages.

"Typical" parameters can and do vary in different applications. All operating parameters, including "Typicals" must be validated for each customer application by the customer's technical experts.

Samsung products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, for other applications intended to support or sustain life, or for any other application in which the failure of the Samsung product could create a situation where personal injury or death may occur.

Should the Buyer purchase or use a Samsung product for any such unintended or unauthorized application, the Buyer shall indemnify and hold Samsung and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, expenses, and reasonable attorney fees arising out of, either directly or indirectly, any claim of personal injury or death that may be associated with such unintended or unauthorized use, even if such claim alleges that Samsung was negligent regarding the design or manufacture of said product.

DREX-1 v2.0, High-Performance SDRAM Controller User's Guide, Revision 2.0

Samsung Electronics Co., Ltd.
San #24 Nongseo-Dong, Giheung-Gu
Yongin-City, Gyeonggi-Do, Korea 446-711

Contact Us: *taehun.kim@samsung.com*
TEL: (82)-(31)-209-9672
FAX: (82)-(31)-209-4494

Home Page: http://www.samsungsemi.com

Printed in the Republic of Korea

Revision History

Revision No.	Date	Description	Refer to	Author(s)

Table of Contents

1 DREX-1 V2.0 1-1

1.1 Overview 1-1
1.1.1 Introduction 1-1
1.1.2 Features 1-1
1.1.3 List of Supported SDRAM Devices 1-1
1.1.4 Design Limitations 1-2
1.2 Architecture Overview 1-3
1.3 Initialization 1-5
1.3.1 LPDDR2/3 1-6
1.3.2 DDR3 1-9
1.4 Address Mapping 1-11
1.4.1 Linear mapping 1-11
1.4.2 Interleaved mapping 1-11
1.5 Low Power Operation 1-13
1.5.1 AXI low power channel 1-13
1.5.2 Dynamic power down 1-13
1.5.3 Dynamic self refresh 1-13
1.5.4 Clock stop 1-13
1.5.5 Direct command 1-13
1.6 Precharge Policy 1-14
1.6.1 Bank Selective Precharge 1-14
1.6.2 Timeout precharge 1-14
1.7 Quality of Service 1-15
1.7.1 BRB Arbitration 1-15
1.7.2 AxQoS based QoS 1-15
1.7.3 BRB Space Reservation 1-15
1.8 Performance Profiling 1-16
1.9 Pause Feature 1-19
1.10 CA swap 1-20
1.11 Register Descriptions 1-21
1.11.1 Register Overview 1-21
1.11.2 Controller Control Register (ConControl, R/W, Address Offset=0x0000) 1-23
1.11.3 Memory Control Register (MemControl, R/W, Address Offset=0x0004) 1-25
1.11.4 Memory Chip0 Configuration Register (MemConfig0, R/W, Address Offset=0x0008) 1-27
1.11.5 Memory Chip1 Configuration Register (MemConfig1, R/W, Address Offset=0x000C) 1-28
1.11.6 Memory Direct Command Register (DirectCmd, R/W, Address Offset=0x0010) 1-29
1.11.7 Precharge Policy Configuration Register (PrechConfig, R/W, Address Offset=0x14) 1-31
1.11.8 PHY Control0 Register (PhyControl0, R/W, Address Offset=0x0018) 1-32
1.11.9 Dynamic Power Down Configuration Register (PwrdnConfig, R/W, Address Offset=0x0028) 1-33
1.11.10 AC Timing Register for Periodic ZQ(ZQCS) of Memory (TimingPZQ, R/W, Address Offset=0x002C) 1-34
1.11.11 AC Timing Register for Auto Refresh of Memory (TimingAref, R/W, Address Offset=0x0030) 1-34
1.11.12 AC Timing Register for the Row of Memory (TimingRow, R/W, Address Offset=0x0034) 1-34
1.11.13 AC Timing Register for the Data of Memory (TimingData, R/W, Address Offset=0x0038) 1-36
1.11.14 AC Timing Register for the Power modes of Memory (TimingPower, R/W, Address Offset=0x003C) 1-38
1.11.15 PHY Status Register (PhyStatus, Read Only, Address Offset=0x0040) 1-39
1.11.16 Memory ChipStatus Register (ChipStatus, Read Only, Address Offset=0x0048) 1-40

1.11.17 Memory Mode Registers Status Register (MrStatus, Read Only, Address Offset=0x0054)..... 1-41
1.11.18 Quality of Service Control Register n (QosControl n, R/W, Address Offset=0x0060 + 8n (n=0~15, integer))..... 1-41
1.11.19 Write Training Configuration Register (WrTraConfig, R/W, Address Offset=0x00F4) 1-42
1.11.20 Read Leveling Configuration Register (RdlvlConfig, R/W, Address Offset=0x00F8) 1-43
1.11.21 BRB Reservation Control Register (BRBRSVCONTROL, R/W, Address Offset=0x0100)..... 1-44
1.11.22 BRB Reservation Configuation Register (BRBRSVCONFIG, R/W, Address Offset=0x0104)... 1-45
1.11.23 BRB QoS Configuation Register (BRBQOSCONFIG, R/W, Address Offset=0x0108) 1-46
1.11.24 Memory Chip0 Base Configuration Register (MemBaseConfig0, R/W, Address Offset=0x010C) 1-47
1.11.25 Memory Chip1 Base Configuration Register (MemBaseConfig1, R/W, Address Offset=0x0110). 1-48
1.11.26 Write Leveling Configuration Register0 (WRLVLCONFIG0, R/W, Address Offset=0x0120)..... 1-49
1.11.27 Write Leveling Configuration Register1 (WRLVLCONFIG1, R/W, Address Offset=0x0124)..... 1-49
1.11.28 Write Leveling Status Register (WRLVLSTATUS, R/W, Address Offset=0x0128)..... 1-49
1.11.29 Performance Event Configuration0 Register (PerevConfig0, R/W, Address Offset=0x0134) ... 1-50
1.11.30 Performance Event Configuration1 Register (PerevConfig1, R/W, Address Offset=0x0138) ... 1-50
1.11.31 Performance Event Configuration2 Register (PerevConfig2, R/W, Address Offset=0x013C)... 1-51
1.11.32 Performance Event3 Configuration Register (PerevConfig3, R/W, Address Offset=0x0140) ... 1-51
1.11.33 CTRL_IO_RDATA Register (CTRL_IO_RDATA, R, Address Offset=0x0150) 1-51
1.11.34 CA Calibration Configuration Register0 (CACAL_CONFIG0, R/W, Address Offset=0x0160)... 1-51
1.11.35 CA Calibration Configuration Register (CACAL_CONFIG1, R/W, Address Offset=0x0164)..... 1-52
1.11.36 CA Calibration Status Register (CACAL_STATUS, R, Address Offset=0x0168) 1-52
1.11.37 Performance Monitor Control Register (PMNC_PPC, R/W, Address Offset=0xE000)..... 1-52
1.11.38 Count Enable Set Register (CNTENS_PPC, R/W, Address Offset=0xE010)..... 1-54
1.11.39 Count Enable Clear Register (CNTENC_PPC, R/W, Address Offset=0xE020) 1-54
1.11.40 Interrupt Enable Set Register (INTENS_PPC, R/W, Address Offset=0xE030)..... 1-54
1.11.41 Interrupt Enable Clear Register (INTENC_PPC, R/W, Address Offset=0xE040) 1-55
1.11.42 Overflow Flag Status Register (FLAG_PPC, R/W, Address Offset=0xE050) 1-55
1.11.43 Cycle Count Register (CCNT_PPC, R/W, Address Offset=0xE100) 1-56
1.11.44 Performance Monitor Count0 Register (PMCNT0_PPC, R/W, Address Offset=0xE110)..... 1-56
1.11.45 Performance Monitor Count1 Register (PMCNT1_PPC, R/W, Address Offset=0xE120)..... 1-56
1.11.46 Performance Monitor Count2 Register (PMCNT2_PPC, R/W, Address Offset=0xE130)..... 1-56
1.11.47 Performance Monitor Count3 Register (PMCNT3_PPC, R/W, Address Offset=0xE140)..... 1-56

List of Figures

Figure 1-1 Overall Block Diagram 1-3
Figure 1-2 Linear Address Mapping 1-11
Figure 1-3 Interleaved Address Mapping 1-11
Figure 1-4 Timing Diagram Of Timeout Precharge 1-14
Figure 1-5 PAUSE_REQ/PAUSE_ACK handshaking 1-19
Figure 1-6 ca_swap=0 and ca_swap=1 1-20

List of Tables

Table 1-1 List of Performance Events 1-16
Table 1-2 List of Performance Information 1-16
Table 1-3 Enable/Disable & Start/Stop Control 1-17

List of Conventions

Register RW Access Type Conventions

Type	Definition	Description
R	Read Only	The application has permission to read the Register field. Writes to read-only fields have no effect.
W	Write Only	The application has permission to write in the Register field.
RW	Read & Write	The application has permission to read and writes in the Register field. The application sets this field by writing 1'b1 and clears it by writing 1'b0.

Register Value Conventions

Expression	Description
x	Undefined bit
X	Undefined multiple bits
?	Undefined, but depends on the device or pin status
Device dependent	The value depends on the device
Pin value	The value depends on the pin status

Reset Value Conventions

Expression	Description
0	
1	
x	

Warning: Some bits of control registers are driven by hardware or write only. As a result the indicated reset value and the read value after reset might be different.

List of Acronyms

Acronyms	Descriptions

1 DREX-1 V2.0

1.1 Overview

1.1.1 Introduction

The DREX-1 v2.0 is an Advanced Microcontroller Bus Architecture (AMBA) AXI-compliant slave to interface external JEDEC standard SDRAM devices. To support high-speed memory devices, the DREX-1 uses the DFI (DDR PHY Interface) as an interface protocol with SDRAM PHY. It features an advanced scheduler in an effort to transfer data to or from memory devices efficiently. Furthermore, its internal pipeline stages are optimized for higher performance and lower latency.

1.1.2 Features

- Compatible with JEDEC standard LPDDR2-S4/LPDDR3/DDR3 SDRAMs
- Supports 1:1 synchronous operation between AXI bus ACLK and scheduler CCLK
- Supports DFI 1:2 synchronous operation between ACLK/CCLK and memory clock domain
- Parameterized number of slave ports (1, 2 3 or 4) compatible with AMBA3 AXI protocol for accesses to SDRAM devices
- One slave port compatible with AMBA3 APB protocol for programmable special function registers
- Uses the DFI SDRAM PHY interface to support high-speed memory devices
- Supports up to two memory ranks (chip selects) and 4/8 banks per memory chips
- Supports 512Mb, 1Gb, 2Gb, 4Gb and 8Gbit density per a chip select
- Supports QoS scheme to ensure low latency for real-time applications
- Out-of order scheduling policy for higher performance
- Detects AXI RAR/WAW hazards automatically
- Supports early write response
- Supports rank/bank interleaving
- Supports AMBA AXI low power interface for systemic power control
- Adapts to various low power schemes to reduce the dynamic and static current of memory
- Supports outstanding exclusive accesses
- Supports bank selective precharge policy
- Accommodates the embedded performance monitor
- ca_swap signal for reversing ca[9:0] to ca[0:9] for LPDDR2/LPDDR3.

1.1.3 List of Supported SDRAM Devices

DREX-1 v2.0 supports the following SDRAM devices:

- LPDDR2-S4
- LPDDR3
- DDR3

1.1.4 Design Limitations

- Does not support locked access on all AMBA3 AXI/APB slave ports.
- Supports only DRAM data channel width of 32 bits.

1.2 Architecture Overview

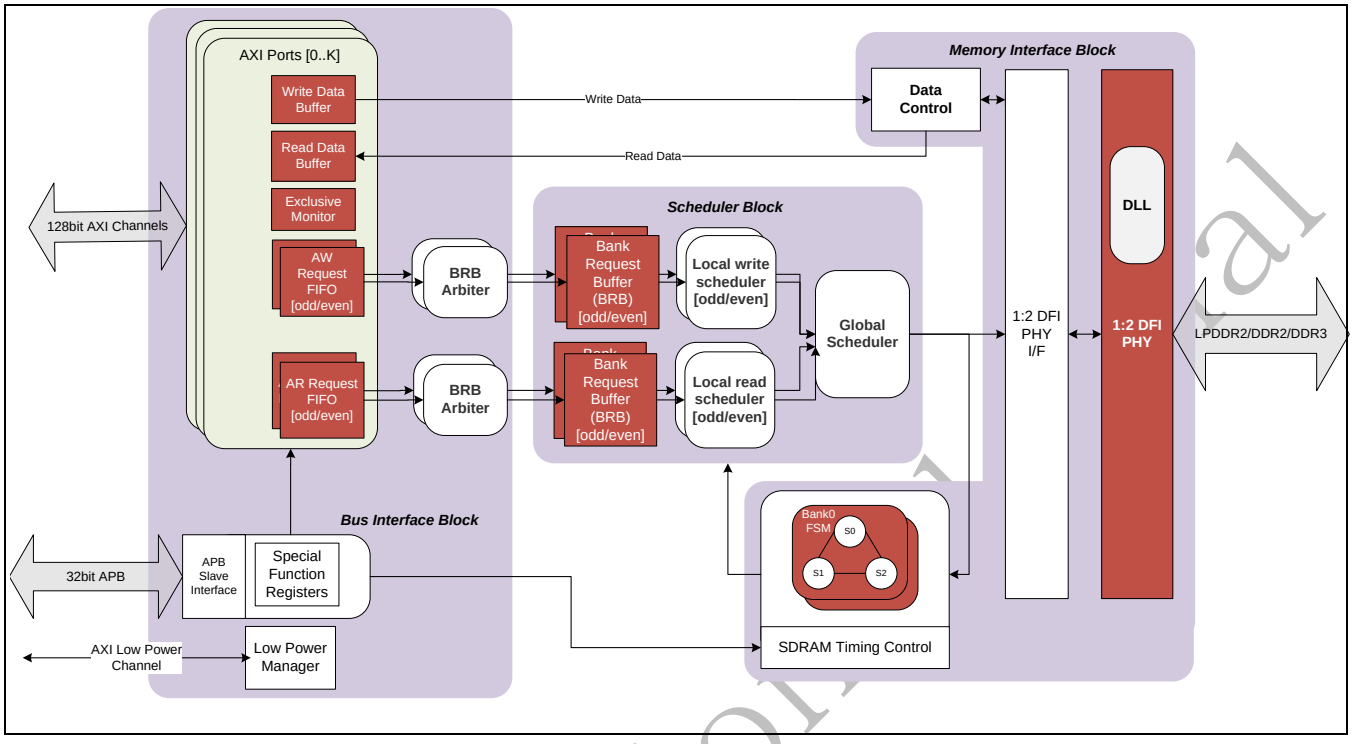

Figure 1-1 Overall Block Diagram

Figure 1-1 shows the overall block diagram of DREX-1. The block diagram shows the Bus Interface block, Scheduler Block, and Memory Interface Block, which connects and interfaces with DFI SDRAM PHYs.

The Bus Interface Block receives AXI transactions for memory access through an AXI slave port and inserts them into the request FIFO1. Meanwhile, it translates the address specified in an AXI transaction into memory address in the form of {rank, bank, row, column} format. Additionally, it stores incoming write data into the Write Data Buffer, whereas it sends read data retained within the Read Data Buffer to a master IP through the AXI slave port. The Read Data Buffer queues data when AXI masters are not ready to accept read data.

The APB slave interface for special function registers/direct commands and an AXI low power channel interface.

The Scheduler Block is where all the out-of-order operations happen. DREX-1 has a distributed scheduler which has 4 local schedulers (2 reads, 2 writes), each with their own request queues (Bank Request Buffers, or BRB), and a global scheduler which coordinates each of the local schedulers' output to generate requests to the PHY. The Scheduler Block users the SDRAM's state information from the Memory Interface Block to determine the optimal schedule for maximum performance. Meanwhile, it translates each memory request in the request buffers into appropriate memory command (e.g. PRE, ACT, RD, RDA, WR, WRA) as required by the corresponding bank state.

The Memory Interface Block translates and maintains the state of the DRAMs using multiple finite state machines (FSM). Additionally, it translates memory commands coming from the Scheduler Block into corresponding SDRAM pin level signals and sends them to the PHY via the PHY Interface. It also updates each memory bank

1 Transactions are distributed to each BRB and FIFO based on the bank number (odd/even) of the access. For example, a read request with a bank number of 3 will be queued on the read 'odd' BRB.

state according to the memory command and sends the bank state back to the scheduler.

1.3 Initialization

SDRAM devices must be powered up and initialized in a predefined manner. Operational procedures other than those specified may result in undefined operation. An initialization procedure consists of three procedures such as PHY DLL initialization, setting controller register and memory initialization. For memory initialization, please refer to JEDEC specifications and datasheets of memory devices. According to the memory types, initialization sequences are as follows.

1.3.1 LPDDR2/3

The following sequence should be used to initialize LPDDR2 devices. Unless specified otherwise, these steps are mandatory.

1. To provide stable power for memory device, DREX-1 must assert and hold CKE to a logic low level. Then apply stable clock.
2. Set the right value to PHY control register0 for LPDDR2/3 operation mode. If read leveling is needed, check LPDDR2/3 IO calibration MRR data and match it to PHY control register1's ctrl_rlvl_rdata_adj field. (Refer to PHY manual)
3. Set the PHY for dqs pulldown mode. (Refer to PHY manual, PHY control register 14)
4. Set the **ConControl**. At this moment, assert the dfi_init_start field to high but the aref_en field should be off.
5. Wait for the **PhyStatus0.dfi_init_complete** field to change to '1'.
6. Set the **ConControl**. At this moment, deassert the dfi_init_start field to low and the aref_en field should be off.
7. Set the **PhyControl0.fp_resync** bit-field to '1' to update DLL information for at least one pclk cycle.
8. Set the **PhyControl0.fp_resync** bit-field to '0'.
9. Set the **MemControl and PhyControl0**. At this moment, all power down modes including sl_dll_dyn_con should be off.
10. Set the **MemBaseConfig0** register. If there are two external memory chips, set the **MemBaseConfig1** register.
11. Set the **MemConfig0** register. If there are two external memory chips, also set the **MemConfig1** register.
12. Set the **PrechConfig** and **PwrdnConfig** registers.
13. Set the **TimingAref, TimingRow, TimingData** and **TimingPower** registers according to memory AC parameters.
14. If QoS scheme is required, set the **QosControl0~15** and **QosConfig0~15** registers.
15. Set the PHY for LPDDR2/3 initialization. LPDDR2/3 initialization clock period is 18ns ~ 100ns. If LPDDR2/3 initialization clock period is 20ns, then follow below steps 16 ~ 22(refer to PHY manual).
16. Set the PHY ctrl_offsetr0~3 and ctrl_offsetw0~3 value to 0x7F
17. Set the PHY ctrl_offsetd value to 0x7F.
18. Set the PHY ctrl_force value to 0x7F.
19. Set the PHY ctrl_dll_on to low.
20. Wait for 10 PCLK cycles.
21. Set the **PhyControl0.fp_resync** bit-field to '1' to update DLL information.
22. Set the **PhyControl0.fp_resync** bit-field to '0'.
23. Confirm that CKE has been as a logic low level at least 100ns after power on
24. Issue a NOP command using the **DirectCmd** register to assert and to hold CKE to a logic high level.
25. Wait for minimum 200us.
26. Issue a MRS command using the **DirectCmd** register to reset memory devices and program the operating parameters.
27. Wait for minimum 10us.
28. Issue proper lpddr2 initialization commands according to specification such as IO calibration (MR #10), device feature1,2 (MR #1, #2). Refer to LPDDR2/3 specification for details.

29. If there are two external memory chips, perform steps 24 ~ 28 for chip1 memory device.
30. Set the PHY ctrl_offsetr0~3 and ctrl_offsetw0~3 value to 0x0
31. Set the PHY ctrl_offsetd value to 0x0
32. Set the PHY ctrl_dll_on enable
33. Wait for 10 PCLK cycles.
34. Set the PHY ctrl_start value to '0'.
35. Set the PHY ctrl_start value to '1'.
36. Wait for 10 PCLK cycles.
37. Wait for the **PhyStatus0.dfi_init_complete** field to change to '1'.
38. Set the **PhyControl0.fp_resync** bit-field to '1' to update DLL information.
39. Set the **PhyControl0.fp_resync** bit-field to '0'.
40. If any leveling/training is needed, disable ctrl_dll_on and set ctrl_force value. (Refer to PHY manual)
41. If write leveling for LPDDR3 is not needed, skip this procedure. If write leveling is needed, set LPDDR3 into write leveling mode using MRS command, set ODT pin high and tWLO using WRLVL_CONFIG0 register(offset=0x120) and set the related PHY SFR fields through PHY APB Interface (Refer to PHY manual). To generate 1 cycle pulse of dfi_wrdata_en_p0, write 0x1 to WRLVL_CONFIG1 register (Offset addr=0x124). To read the value of memory data, use CTRL_IO_RDATA(offset = 0x150). If write leveling is finished, then set ODT pin low and disable write leveling mode of LPDDR3
42. If CA calibration for LPDDR3 is not needed, skip this procedure. If CA calibration is needed, set the related PHY SFR fields through PHY APB Interface (Refer to PHY manual). Set LPDDR3 into CA calibration mode through MR41. For CKE assertion, deassetion, CA value and tADDR setting, use CACAL_CONFIG0(offset = 0x160). For Generation 1 cycle pulse of dfi_csn_p0, use CACAL_CONFIG1(offset = 0x164). To read the value of memory data, use CTRL_IO_RDATA_CH0/CH1 (offset = 0x150, 0x154). Note that CKE pin should be asserted when MR41/48/42 command is issued and CKE pin should be deasserted during CA calibration. Also note that because CA SDLL code updating needs some cycles to complete, 10 PCLK cycles are needed before issuing next command.
43. If read leveling is not needed, skip 44 ~ 48 and set proper value to PHY control register #2. If read leveling is needed, set the related PHY SFR fields through PHY APB Interface. The related register is PHY control register #1 and #2(mpr value, ctrl_rdlvl_gate_en and ctrl_rdlvl_en register, refer to PHY manual).
44. Set the **RdlvlConfig.ctrl_rdlvl_data_en** bit-field to 1'b1. Gate training is not supported.
45. Wait for the **PhyStatus0.read_level_complete** field to change to '1'.
46. Disable the RdlvlConfig.ctrl_rdlvl_gate_en and ctrl_rdlvl_data_en.
47. Set the **PhyControl0.fp_resync** bit-field to '1' to update DLL information.
48. Set the **PhyControl0.fp_resync** bit-field to '0'.
49. If write training is not needed, skip 50 ~ 55. If write training is nedded, set the related PHY SFR fields through PHY APB Interface. The related register is PHY control register #2 and #26, refer to PHY manual)
50. Set write latency of PHY control register #26.
51. Enable **WrtraConfig.write_training_en** to issue ACT command. Refer to this register definition for row and bank address.
52. Wait for 10 PCLK cycles.
53. Enable write de-skewing of PHY control register #2.
54. Wait for the **PhyStatus0.read_level_complete** field to change to '1'.
55. Disable write de-skewing of PHY control register #2.
56. After all leveling/training are completed, enable ctrl_dll_on. (Refer to PHY manual)

57. Set the **PhyControl0.fp_resync** bit-field to '1' to update DLL information.
58. Set the **PhyControl0.fp_resync** bit-field to '0'.
59. Disable PHY gating control through PHY APB Interface (ctrl_atgate, see PHY manual).
60. If power down modes are required, set the **MemControl** register.
61. Set the **ConControl** to turn on an auto refresh counter.

1.3.2 DDR3

The following sequence should be used to initialize DDR3 devices. Unless specified otherwise, these steps are mandatory.

1. Apply power. RESET# pin of memory needs to be maintained for minimum 200us with stable power. CKE is pulled “Low” anytime before RESET# being de-asserted (min. time 10ns)
2. Set the PHY for DDR3 operation mode. . If read leveling is needed, check DDR3 MPR data and match it to PHY control register1's ctrl_rlvl_rdata_adj field. (Refer to PHY manual)
3. Set the PHY for dqs pulldown mode. (Refer to PHY manual, PHY control register 14)
4. If on die termination is required, enable **PhyControl0.mem_term_en**, **PhyControl0.phy_term_en**.
5. Set the **ConControl**. At this moment, assert the dfi_init_start field to high but the aref_en field should be off.
6. Wait for the **PhyStatus0.dfi_init_complete** field to change to ‘1’.
7. Set the **ConControl**. At this moment, deassert the dfi_init_start field to low and the aref_en field should be off.
8. Set the **PhyControl0.fp_resync** bit-field to ‘1’ to update DLL information.
9. Set the **PhyControl0.fp_resync** bit-field to ‘0’.
10. Set the **MemControl and PhyControl0**. At this moment, all power down modes including sl_dll_dyn_con and periodic ZQ(pzq_en) should be off.
11. Set the **MemBaseConfig0** register. If there are two external memory chips, set the **MemBaseConfig1** register.
12. Set the **MemConfig0** register. If there are two external memory chips, also set the **MemConfig1** register.
13. Set the **PrechConfig** and **PwrdnConfig** registers.
14. Set the **TimingAref**, **TimingRow**, **TimingData** and **TimingPower** registers according to memory AC parameters.
15. If QoS scheme is required, set the **QosControl0~15** and **QosConfig0~15** registers.
16. Confirm that after RESET# is de-asserted, 500 us have passed before CKE becomes active.
17. Confirm that clocks(CK, CK#) need to be started and stabilized for at least 10 ns or 5 tCK (which is larger) before CKE goes active.
18. Issue a NOP command using the **DirectCmd** register to assert and to hold CKE to a logic high level.
19. Wait for tXPR(max(5nCK,tRFC(min)+10ns)) or set tXP to tXPR value before step 17. If the system set tXP to tXPR, then the system must set tXP to proper value before normal memory operation.
20. Issue an EMRS2 command using the **DirectCmd** register to program the operating parameters. Dynamic ODT should be disabled. A10 and A9 should be low.
21. Issue an EMRS3 command using the **DirectCmd** register to program the operating parameters.
22. Issue an EMRS command using the **DirectCmd** register to enable the memory DLL.
23. Issue a MRS command using the **DirectCmd** register to reset the memory DLL.
24. Issues a MRS command using the **DirectCmd** register to program the operating parameters without resetting the memory DLL.
25. Issues a ZQINIT commands using the **DirectCmd** register.
26. If there are two external memory chips, perform steps 18 ~ 25 procedures for chip1 memory device.
27. If any leveling/training is needed, disable ctrl_dll_on and set ctrl_force value. (Refer to PHY manual)
28. If write leveling is not needed, skip this procedure. If write leveling is needed, set DDR3 into write leveling mode using MRS command, set ODT pin high and tWLO using WRLVL_CONFIG0 register(offset=0x120) and

set the related PHY SFR fields through PHY APB Interface (Refer to PHY manual). To generate 1 cycle pulse of dfi_wrdata_en_p0, write 0x1 to WRLVL_CONFIG1 register (Offset addr=0x124). If write leveling is finished, then set ODT pin low and disable write leveling mode of DDR3

29. If gate leveling is not needed, skip 30 ~ 33. Gate leveling is only supported DDR3 over 667MHz. If gate leveling is needed, set the related PHY SFR fields through PHY APB Interface. The related register is PHY control register #0, #1 and #2 (Refer to PHY manual)
30. Set the **RdlvlConfig.ctrl_rdlvl_gate_en** bit-field to 1'b1.
31. Wait for the **PhyStatus0.read_level_complete** field to change to '1'.
32. Disable DQS pulldown mode.(Refer to PHY manual)
33. Disable the RdlvlConfig.ctrl_rdlvl_gate_en and ctrl_rdlvl_data_en.
34. If read leveling is not needed, skip 35 ~ 39 and set proper value to PHY control register #2. If read leveling is needed, set the related PHY SFR fields through PHY APB Interface. The related register is PHY control register #1 and #2 (mpr value, ctrl_rdlvl_gate_en and ctrl_rdlvl_en register, refer to PHY manual).
35. Set the **ctrl_rdlvl_data_en** bit-field to 1'b1.
36. Wait for the **PhyStatus0.read_level_complete** field to change to '1'.
37. Disable the RdlvlConfig.ctrl_rdlvl_gate_en and ctrl_rdlvl_data_en.
38. Set the **PhyControl0.fp_resync** bit-field to '1' to update DLL information.
39. Set the **PhyControl0.fp_resync** bit-field to '0'.
40. If write training is not needed, skip 41 ~ 46. If write training is needed, set the related PHY SFR fields through PHY APB Interface. The related register is PHY control register #2 and #26, refer to PHY manual)
41. Set write latency of PHY control register #26.
42. Enable **WrtraConfig.write_training_en** to issue ACT command. Refer to this register definition for row and bank address.
43. Wait for 10 PCLK cycles.
44. Enable write de-skewing of PHY control register #2.
45. Wait for the **PhyStatus0.read_level_complete** field to change to '1'.
46. Disable write de-skewing of PHY control register #2.
47. After all leveling/training are completed, enable ctrl_dll_on. (Refer to PHY manual)
48. Set the **PhyControl0.fp_resync** bit-field to '1' to update DLL information.
49. Set the **PhyControl0.fp_resync** bit-field to '0'.
50. Disable PHY gating control through PHY APB Interface (ctrl_atgate, refer to PHY manual).
51. If power down modes or periodic ZQ (pzq_en) are required, set the **MemControl** register.
52. Set the **ConControl** to turn on an auto refresh counter.

1.4 Address Mapping

DREX-1 modifies the address of the AXI transaction coming from the AXI slave port into a memory address – chip select, bank address, row address, column address and memory data width.

To map chip select of memory device to a specific area of the address map, the **chip_base** and **chip_mask** bit-fields of the **MemConfig0** register needs to be set (Refer to Register Descriptions). If chip1 of the memory device exists, the **MemConfig1** register must also be set. Then, the AXI address requested by AXI Masters is divided into the AXI base address and AXI offset address. The AXI base address activates the appropriate memory chip select and the AXI offset address is mapped to a memory address according to the bank, row, column number, and data width set by the **MemConfig0/1** and **MemControl** register.

There are two ways to map the AXI offset address as shown below: 1) linear mapping 2) interleaved mapping.

1.4.1 Linear mapping

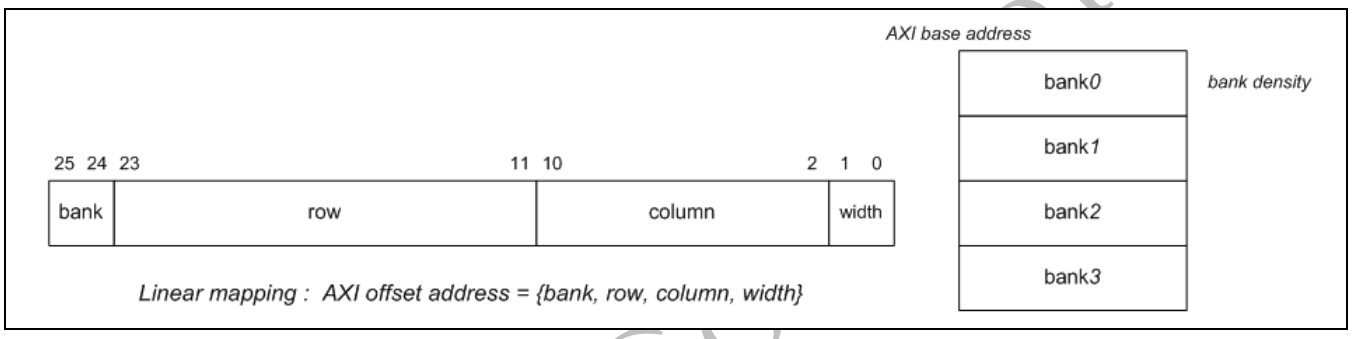

Figure 1-2 Linear Address Mapping

As shown in *Figure 1-2* the linear mapping method maps the AXI address in the order of bank, row, column and width3. Since the bank address does not change for at least one bank size, applications that use linear address mapping have a high possibility to access the same bank.

1.4.2 Interleaved mapping

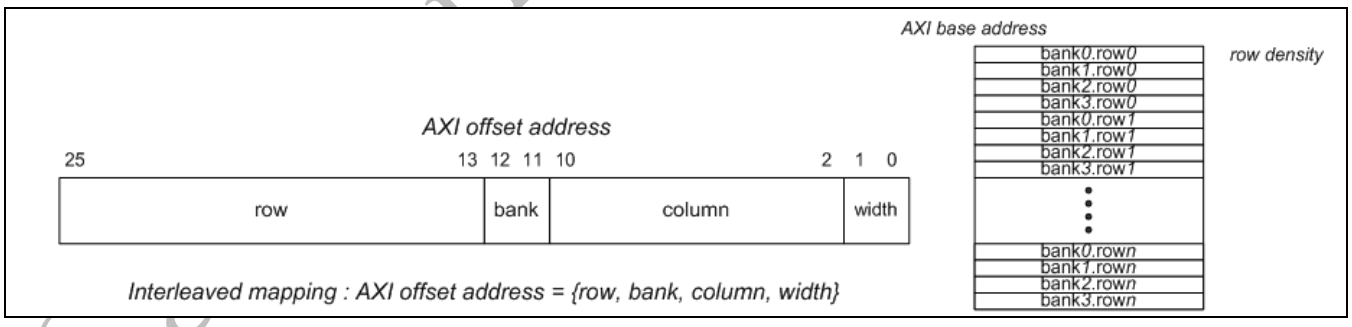

Figure 1-3 Interleaved Address Mapping

As shown in *Figure 1-3* the interleaved mapping method maps the AXI address in the order of row, bank, column and width. The difference between above two methods is that the bank and row order is different. For accesses

3 'width' represents the data width of the DRAM used, which is fixed to 32 bits(4 bytes), and hence, on a byte-addressed address value, the corresponding width is fixed to 2.

beyond a row size, interleaved mapping accesses a different bank. Therefore, applications that use interleaved mapping access numerous banks. It causes better performance but more power consumption.

1.5 Low Power Operation

The controller executes a low power memory operation in five ways as described below. Each feature is independent of each other and executed at the same time.

1.5.1 AXI low power channel

The controller has an AXI low power channel interface to communicate with low power management units such as the system controller, which makes the memory device go into self refresh mode.

1.5.2 Dynamic power down

An SDRAM device has an active/precharge power down mode. This mode is triggered by deasserting CKE to LOW. When any of the banks is open, it enters active power down mode. Otherwise, it enters precharge power down mode.

When the request buffers remain empty for certain number of cycles (PwrdnConfig.dpwrdn_cyc register), DREX-1 changes the memory device's state to active/precharge power down automatically. The memory device enters Active/precharge power down mode or Forced precharge power down mode according to the SFR setting. The description of the two power down modes are as follows:

1. Active/precharge power down mode: Enter power down w/o considering whether there is a row open or not.
2. Forced precharge power down mode: Enter power down after closing all banks.

When DREX-1 receives a new AXI transaction while memory device is in power down mode, it automatically wakes up the memory device from power down state and executes in a normal operation state.

1.5.3 Dynamic self refresh

Similarly to the dynamic power down feature, if the request buffers remain empty for certain number of cycles (PwrdnConfig.dsref_cyc register), DREX-1 changes the memory device's state to self-refresh mode. Since exiting power down mode requires many cycles, a longer idle cycle threshold is recommended for dynamic self-refresh entry than the threshold for dynamic power down.

1.5.4 Clock stop

To reduce the I/O power of the memory device and the controller, it is possible to stop the clock if the LPDDR / LPDDR2-S4 is in idle mode, or self refresh mode and DDR2/DDR3 is in self refresh mode. If this feature is enabled, the controller automatically executes the clock stop feature. In DDR3, clock stop feature must be turn-on and off considering tCKSRX/tCKSRE/tCKESR timing by software.

1.5.5 Direct command

Use the direct command feature to send a memory command directly to the memory device through the APB3 port. This way, it is possible to force the memory device to enter active/precharge power down, self-refresh or deep power down mode.

1.6 Precharge Policy

There are two options for DREX-1 regarding precharge policy – bank-selective precharge and timeout precharge.

1.6.1 Bank Selective Precharge

Since applications have different page policy preferences, it is hard for the engineer to decide on whether to use open page policy, or close page (auto precharge) policy. Instead of applying the page policy to all of the banks, the bank selective precharge policy allows the user to choose a precharge policy for each bank (refer to **PrechConfig.chip1_policy**). This way, you assign certain applications to a bank that uses an open page policy, and other applications to a bank that uses a close page (auto precharge) policy.

- Open Page Policy: After a READ or WRITE, the accessed row is left open.

- Close Page (Auto Precharge) Policy: When DREX-1 issues the last READ or WRITE CAS command, it augments the command with an auto precharge flag.

1.6.2 Timeout precharge

If a certain bank uses an open page policy, the row is left open after a data access. If this happens and the bank that is left open is not scheduled for a specific number of cycles (**PrechConfig.tp_cnt** bit-field) the controller automatically issues a precharge command to close the bank.

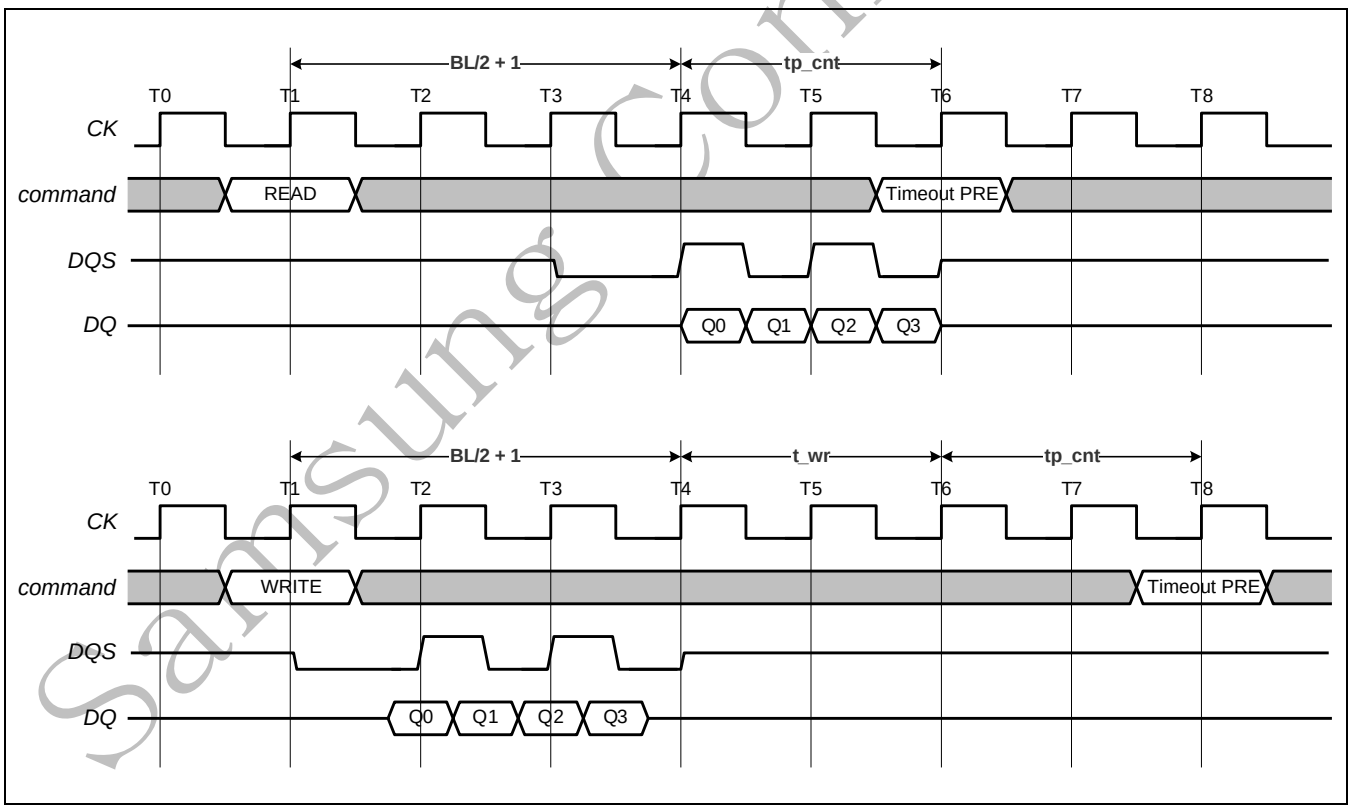

Figure 1-4 Timing Diagram Of Timeout Precharge

1.7 Quality of Service

DREX-1 provides Quality of Service (QoS) feature to ensure low latency for real-time masters. Specifically, DREX-1 uses timeout based QoS enforcement scheme. When DREX-1 receives an AXI transaction, a predefined QoS timeout value is assigned to the corresponding memory request for timeout. When the timer expires, the request gets promoted to the highest priority for immediate selection during arbitration stage.

1.7.1 BRB Arbitration

The AXI requests of the head of AXI Request FIFO in the ports are arbitrated to Bank Request Buffer (BRB). The priority of each port is determined by the highest AxQoS value of the AXI requests in the FIFO. When the AXI Request FIFO timer (**ConControl.timeout_level0)** expires, the port of the FIFO has highest priority to prevent starvation of low priority port.

1.7.2 AxQoS based QoS

When DREX-1 receives an AXI request, it assigns a predefined BRB timeout value (**QosControl_n.cfg_qos, n= 0 ~ 15**) according to the AxQoS value (0 ~ 15). When the BRB timer expires, the request has the highest priority for the Request Scheduler. When the BRB is full or the data buffer for the request is full, the BRB timer decrements by predefined value (**BRBQosConfig.brb_qos_timer_dec**) instead of 1.

1.7.3 BRB Space Reservation

When BRB space reservation is enabled for a AXI AR/AW port (**BRBRsvControl.brb_rsv_en_{w,r}{0,1,2,3}**), the AXI port stops issuing request for BRB arbitration when the occupancy of the target BRB exceeds BRB threshold value (**BRBRsvConfig.brb_rsv_th_{w,r}{0,1,2,3}**). This feature can be used for reserving a BRB slot for slave ports with latency-critical traffic, so that they do not need to be waiting on the AR/AW request FIFO.

1.8 Performance Profiling

DREX-1 provides performance monitoring capability based on event counters accessible through APB interface. Relevant registers are located on 0xFC, 0xE000 ~ 0xE140.

DREX-1 has event counters which is called PPC. Table 1-1 lists performance events in each domain.

It is possible to extract useful information from these event counts. Table 1-2 shows some examples.

Table 1-1 List of Performance Events

Event Items	Clock Domain
1. Total read requests counts	aclk
2.Total read requests counts per bank	aclk
3. Total write requests counts	aclk
4.Total write requests counts per bank	aclk
5 Interleaving, channel0 only and channel1 only access counts	cclk
6. Cas command scheduled counts	cclk
7. Page miss counts	cclk
8.DRAM data channel cycle used	cclk

Table 1-2 List of Performance Information

Information	Event Expression
Total read requests	1
Total write requests	2
Total requests	1+2
Page miss	7
Memory Data Transfer	11

For example, page_hit can be obtained like below. PPC is needed to be read.

Step1: Initialize State

Mode Selection: set perev0~3_sel to 0x68~0x6b. Then 4-bit events for PPC is {cas scheduled on channel1, cas scheduled on channel0, page miss on channel1, page miss on channel0}.

Interrupt Enable: set INTENS_PPC(Offset=0xE030) to 0x8000_000F

Counter Enable: set CNTENS_PPC(Offset=0xE010) to 0x8000_000F

Step2: Start State

Clear Overflow Flag Register: set FLAG_PPC(Offset=0xE050) to 0x8000_000F

All Counter Reset: set PMNC_PPC(Offset=0xE000) to 0x0000_0006

Sampling Duration Initial Value Setup: set CCNT_PPC(Offset=0xE100) to some value

If 10000(0x2710) cycles simulation run, then 0xFFFF_D8EF(0xFFFF_FFFF – 0x2710) need to be set

Start All Counters: set PMNC_PPC(Offset=0xE000) to 0x0000_0001

Step3: Running & Interrupt

Interrupt will be generated from PPC due to overflow on CCNT_PPC

Step4: Stop State

Stop All Counters: set PMNC_PPC(Offset=0xE000) to 0x0000_0000

Step5: Read State

Get Overflow Flag Register: read FLAG_PPC(Offset=0xE050) value to check if interrupt has been generated by CCNT_PPC

Get Performance Counter Value: read PMCNT0~3_PPC for total cas scheduled count number and page miss count number.

Table 1-3 Enable/Disable & Start/Stop Control

Control Function		Description
Enable/Disable [counter/adder]		CNTENS/CNTENC
Enable/Disable [counter/adder]'s interrupt		INTENS/INTENC
Start/Stop PPC	Start_mode == 0	Start & Stop by Register : PMNC[0] == 1 → Start, PMNC[0] == 0 → Stop
	Start_mode == 1	Start & Stop by External Trigger Trigger == 1 → Start (In this case, PMNC[0] read value becomes 1) Trigger == 0 → Stop(In this case, PMNC[0] read value becomes 0)

CNTENC & CNTENS

PPMU_CTENC and PPMU_CTENS are a pair of related registers.

If you want to enable one counter, you should write 1 to its corresponding bit of PPMU_CTENS. After this writing, the values you read from PPMU_CTENC and PPMU_CTENS are both changed, which show the enabled counter's corresponding bit with 1.

If you want to disable one counter, you should write 1 to its corresponding bit of PPMU_CTENC. Still after this writing, the values you read from PPMU_CTENC and PPMU_CTENS are both changed, which show the disabled counter's corresponding bit with 0.

Their initial values are 0x0000_0000.

INTENC & INTENS

PPMU_INTENS and PPMU_INTENC are used for enabling and disabling the interrupt generation of the counters. Their setting rules are same as the two registers above.

CCNT & PMCNTx

PPMU_CCNT is an r/w register.

Before start counting, you can set PPMU_CCNT's initial value by writing some value to it. Read after this write should be the value as the same as the written value.

After starting counting (suppose that CCNT counter is enabled), the PPMU_CCNT will be increased one by every cycle from its initial value until you stop counting. Any value read or written from/to the counter registers during the counting is meaningless.

1.9 Pause Feature

DREX-1 supports pause feature through external ports called "PAUSE_REQ" and "PAUSE_ACK". When PAUSE_ACK is set to low, DREX-1 guarantees that there will be no requests issued to the DRAM until the external port PAUSE_REQ is driven to high. This feature is used for switching the clock frequency of the memory interface.

Figure 1-5 shows the handshaking mechanism of this feature.

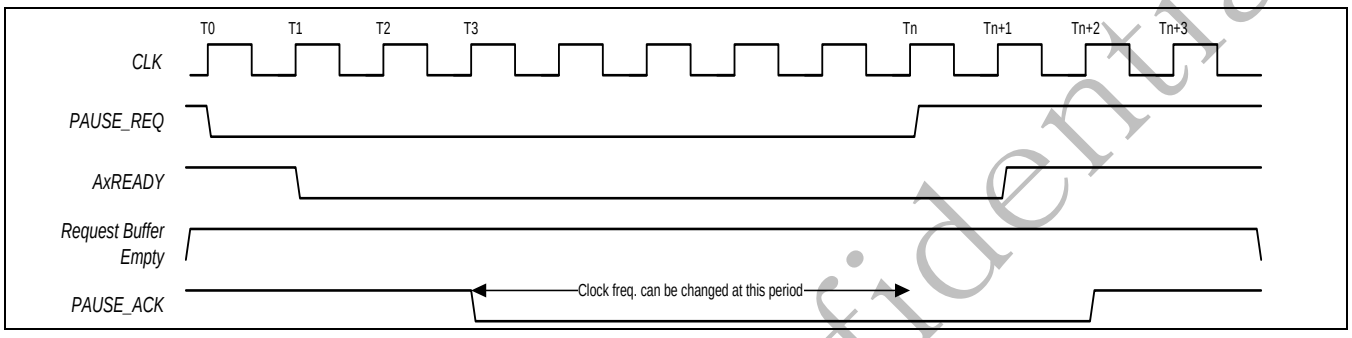

Figure 1-5 PAUSE_REQ/PAUSE_ACK handshaking

Clock frequency change of memory can be applied through this pause feature like below procedures.

1. Pause request setting PAUSE_REQ to low.
2. DREX-1 sets AxREADY to low.
3. DREX-1 finishes memory access until queue empty.
4. DREX-1 sets PAUSE_ACK to low.
5. Clock frequency changes.
6. Release pause request setting PAUSE_REQ to high.
7. DREX-1 sets AxREADY and PAUSE_ACK to high.

Master side of this protocol should not change PAUSE_REQ value before finishing previous handshaking. It means that PAUSE_REQ should be waiting for the PAUSE_ACK before change its value.

1.10 CA swap

DREX-1 has an input signal named ca_swap. If this signal is driven to 1, the DFI interface's address signals will have its bit locations reversed. (addr[9] and addr[0] will be swapped, addr[8] and addr[1] will be swapped, etc) The purpose of this signal is for supporting different packaging solutions, so that the system level designer can choose among one of the modes depending on the routing requirement of the packaging or the PCB using the SoC with DREX-1. Please take care since driving the wrong ca_swap value may render the whole SoC unusable!

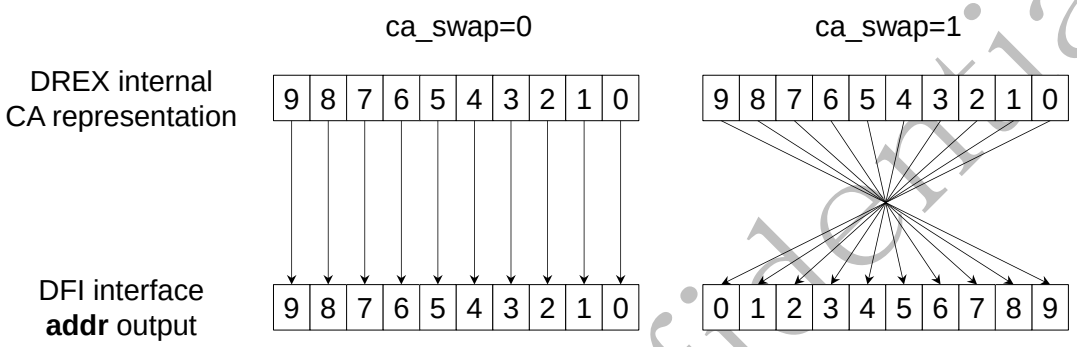

Figure 1-6 ca_swap=0 and ca_swap=1

The ca_swap feature is only valid on LPDDR2/LPDDR3 modes, and has no effect on DDR3 modes.

1.11 Register Descriptions

1.11.1 Register Overview

Register	Offset	R/W	Description	Initial Value
General Registers				
CONCONTROL	0x0000	R/W	Controller Control Register	0x0FFF_1100
MEMCONTROL	0x0004	R/W	Memory Control Register	0x0020_2601
MEMCONFIG0	0x0008	R/W	Memory Chip0 Configuration Register	0x0000_1312
MEMCONFIG1	0x000C	R/W	Memory Chip1 Configuration Register	0x0000_1312
DIRECTCMD	0x0010	R/W	Memory Direct Command Register	0x0000_0000
PRECHCONFIG	0x0014	R/W	Precharge Policy Configuration Register	0xFF00_0000
PHYCONTROL0	0x0018	R/W	PHY Control0 Register	0x0000_0000
PWRDNCONFIG	0x0028	R/W	Dynamic Power Down Configuration Register	0xFFFF_00FF
TIMINGPZQ	0x002C	R/W	AC Timing Register for DDR3 SDRAM periodic ZQ(ZQCS)	0x0000_4084
TIMINGAREF	0x0030	R/W	AC Timing Register for SDRAM Auto refresh	0x0000_005D
TIMINGROW	0x0034	R/W	AC Timing Register for SDRAM Row	0x1F23_3286
TIMINGDATA	0x0038	R/W	AC Timing Register for SDRAM Data	0x1230_360C
TIMINGPOWER	0x003C	R/W	AC Timing Register for Power Mode of SDRAM	0x381B_0422
PHYSTATUS	0x0040	R	PHY Status Register	0x0000_0000
CHIPSTATUS	0x0048	R	Memory Status Register	0x0000_0000
MRSTATUS	0x0054	R	Memory Mode Registers Status Register	0x0000_0000
QOSCONTROL0	0x0060	R/W	Quality of Service Control Register 0	0x0000_0FFF
QOSCONTROL1	0x0068	R/W	Quality of Service Control Register 1	0x0000_0FFF
QOSCONTROL2	0x0070	R/W	Quality of Service Control Register 2	0x0000_0FFF
QOSCONTROL3	0x0078	R/W	Quality of Service Control Register 3	0x0000_0FFF
QOSCONTROL4	0x0080	R/W	Quality of Service Control Register 4	0x0000_0FFF
QOSCONTROL5	0x0088	R/W	Quality of Service Control Register 5	0x0000_0FFF
QOSCONTROL6	0x0090	R/W	Quality of Service Control Register 6	0x0000_0FFF
QOSCONTROL7	0x0098	R/W	Quality of Service Control Register 7	0x0000_0FFF
QOSCONTROL8	0x00A0	R/W	Quality of Service Control Register 8	0x0000_0FFF
QOSCONTROL9	0x00A8	R/W	Quality of Service Control Register 9	0x0000_0FFF
QOSCONTROL10	0x00B0	R/W	Quality of Service Control Register 10	0x0000_0FFF
QOSCONTROL11	0x00B8	R/W	Quality of Service Control Register 11	0x0000_0FFF
QOSCONTROL12	0x00C0	R/W	Quality of Service Control Register 12	0x0000_0FFF
QOSCONTROL13	0x00C8	R/W	Quality of Service Control Register 13	0x0000_0FFF
QOSCONTROL14	0x00D0	R/W	Quality of Service Control Register 14	0x0000_0FFF

QOSCONTROL15	0x00D8	R/W	Quality of Service Control Register 15	0x0000_0FFF
WRTRA_CONFIG	0x00F4	R/W	Write Training Configuration Register	0x0000_0000
RDLVL_CONFIG	0x00F8	R/W	Read Leveling Configuration Register	0x0000_0000
BRBRSVCONTROL	0x0100	R/W	BRB Reservation Control Register	0x0000_0000
BRBRSVCONFIG	0x0104	R/W	BRB Reservation Configuration Register	0x8888_8888
BRBQOSCONFIG	0x0108	R/W	BRB QoS Configuration Register	0x0000_0010
MEMBASECONFIG0	0x010C	R/W	Memory Chip0 Base Configuration Register	0x0020_07F8
MEMBASECONFIG1	0x0110	R/W	Memory Chip1 Base Configuration Register	0x0028_07F8
WRLVL_CONFIG0	0x0120	R/W	Write Leveling Configuration Register 0	0x0000_0010
WRLVL_CONFIG1	0x0124	R/W	Write Leveling Configuration Register 1	0x0000_0000
WRLVL_STATUS	0x0128	R	Write Leveling Status Register	0x0000_0000
PEREVCONFIG0	0x0134	R/W	Performance Event Configuration0 Register	0x0000_0000
PEREVCONFIG1	0x0138	R/W	Performance Event Configuration1 Register	0x0000_0000
PEREVCONFIG2	0x013C	R/W	Performance Event Configuration2 Register	0x0000_0000
PEREVCONFIG3	0x0140	R/W	Performance Event Configuration3 Register	0x0000_0000
CTRL_IO_RDATA	0x0150	R	CTRL_IO_RDATA from PHY	0x0000_0000
CACAL_CONFIG0	0x0160	R/W	CA Calibration Configuration Register 0	0x003F_F010
CACAL_CONFIG1	0x0164	R/W	CA Calibration Configuration Register 1	0x0000_0000
CACAL_STATUS	0x0168	R	CA Calibration Status Register	0x0000_0000
PMNC_PPC	0xE000	R/W	Performance Monitor Control Register	0x0000_0000
CNTENS_PPC	0xE010	R/W	Count Enable Set Register	0x0000_0000
CNTENC_PPC	0xE020	R/W	Count Enable Clear Register	0x0000_0000
INTENS_PPC	0xE030	R/W	Interrupt Enable Set Register	0x0000_0000
INTENC_PPC	0xE040	R/W	Interrupt Enable Clear Register	0x0000_0000
FLAG_PPC	0xE050	R/W	Overflow Flag Status Register	0x0000_0000
CCNT_PPC	0xE100	R/W	Cycle Count Register	0x0000_0000
PMCNT0_PPC	0xE110	R/W	Performance Monitor Count Register	0x0000_0000
PMCNT1_PPC	0xE120	R/W	Performance Monitor Count Register	0x0000_0000
PMCNT2_PPC	0xE130	R/W	Performance Monitor Count Register	0x0000_0000
PMCNT3_PPC	0xE140	R/W	Performance Monitor Count Register	0x0000_0000

1.11.2 Controller Control Register (ConControl, R/W, Address Offset=0x0000)

Field	Bit	Description	R/W	Initial State
Reserved	[31:29]	Should be zero		0x0
dfi_init_start	[28]	DFI PHY initialization start This field is used to initialize DFI PHY. Set this field to 1 to initialize DFI PHY and set this field to 0 after received dfi_init_complete of PhyStatus register.	R/W	0x0
timeout_level0	[27:16]	**Default Timeout Cycles** This counter prevents transactions in the AXI request FIFO from starvation. This counter starts if a new AXI transaction comes into the request FIFO. If the counter becomes zero, the corresponding FIFO has the highest priority during BRB arbitration. Refer to chapter 1.7 . Quality of Service for detailed information	R/W	0xFFF
Reserved	[15]	Should be zero		0x0
rd_fetch	[14:12]	**Read Data Fetch Cycles** 0xn = n cclk cycles (cclk : DREX-1 core clock) The recommended value of this field is 0x2 for LPDDR3 800MHz memory clock and other cases are 0x1. This register is for the unpredictable latency of read data coming from memory devices by tDQSCK variation or the board flying time. The read fetch delay of PHY read FIFO must be controlled by this parameter. The controller will fetch read data from PHY after read_latency + n cclk cycles. Refer to chapter 1.3 for detailed information.	R/W	0x1
Reserved	[11:9]	Should be zero		0x0
empty	[8]	**Empty Status** 0x0 = Not Empty, **0x1 = Empty** There is no AXI transaction in memory controller.	R	0x1
Reserved	[7:6]	Should be zero		0x0
aref_en	[5]	**Auto Refresh Counter** 0x0 = Disable, 0x1 = Enable Enable this to decrease the auto refresh counter by 1 at the rising edge of the rclk	R/W	0x0
Reserved	[4]	Should be zero		0x0
io_pd_con	[3]	**I/O Powerdown Control in Low Power Mode(through LPI)** 0x0 = Use programmed ctrl_pd and pulldown control 0x1 = Automatic control for ctrl_pd and pulldown control If this value is set to 0x1 and in low power mode through LPI,	R/W	0x0

		DREX-1 automatically sets powerdown enable for input buffer of I/O and pulldown disable for dq and dqs in powerdown mode If this value is set to 0x0, DREX-1 only sends programmed ctrl_pd value and pulldown control.		
clk_ratio	[2:1]	**Clock Ratio of Bus Clock to Memory Clock** 0x0 = freq.(aclk): freq.(cclk) = 1: 1, 0x1 ~ 0x3 = Reserved	R/W	0x0
Reserved	[0]	Should be zero		0x0

1.11.3 Memory Control Register (MemControl, R/W, Address Offset=0x0004)

Field	Bit	Description	R/W	Initial State
Reserved	[31:27]	Should be zero		0x0
mrr_byte	[26:25]	**Mode Register Read byte lane location** 0x0 = memory dq[7:0] 0x1 = memory dq[15:8] 0x2 = memory dq[23:16] 0x3 = memory dq[31:24]	R/W	0x0
pzq_en	[24]	DDR3 periodic ZQ(ZQCS) enable Note that after exit from self refresh, ZQ function is required by the software. The controller does not support ZQ calibration command to be issued in parallel to DLL lock time when coming out of self refresh. Turn-on only when using DDR3. The periodic ZQ interval is defined by t_pzq in TIMINGPZQ register.	R/W	0x0
Reserved	[23]	Should be zero		0x0
bl	[22:20]	**Memory Burst Length** 0x0~1 = Reserved, 0x2 = 4, 0x3 = 8, 0x4 ~ 0x7 = Reserved In case of LPDDR2-S4, the controller only supports burst length 4. In case of DDR3 and LPDDR3, the controller only supports burst length 8.	R/W	0x2
num_chip	[19:16]	**Number of Memory Chips** 0x0 = 1 chip, 0x1 = 2 chips, 0x2 ~ 0xf = Reserved	R/W	0x0
mem_width	[15:12]	**Width of Memory Data Bus** 0x0 ~ 0x1 = Reserved, 0x2 = 32-bit, 0x3 ~ 0xf= Reserved	R/W	0x2
mem_type	[11:8]	Type of Memory 0x0 ~ 0x4 = Reserved, 0x5 = LPDDR2-S4, 0x6 = DDR3, 0x7 = LPDDR3, 0x8 ~ 0xf = Reserved	R/W	0x6
add_lat_pall	[7:6]	**Additional Latency for PALL in cclk cycle** 0x0 = 0 cycle, 0x1 = 1 cycle 0x2 = 2 cycle,	R/W	0x0

		0x3 = Reserved If all banks precharge command is issued, the latency of pre-charging will be tRP + add_lat_pall		
dsref_en	[5]	**Dynamic Self Refresh** 0x0 = Disable, 0x1 = Enable Refer to chapter 1.5.2 . Dynamic power down for detailed information. In DDR3, this feature is not supported. This feature must be turn-off when using DDR3.	R/W	0x0
tp_en	[4]	**Timeout Precharge** 0x0 = Disable, 0x1 = Enable If tp_en is enabled, it automatically precharges an open bank after a specified amount of mclk cycles (if no access has been made in between the cycles) in an open page policy. If **PrechConfig.tp_cnt** bit-field is set, it specifies the amount of mclk cycles to wait until timeout precharge precharges the open bank. Refer to chapter 1.6.2 . Timeout precharge for detailed information.	R/W	0x0
dpwrdn_type	[3:2]	**Type of Dynamic Power Down** 0x0 = Active/precharge power down, 0x1 = Forced precharge power down 0x2 ~ 0x3 = Reserved Refer to chapter 1.5.2 . Dynamic power down for detailed information.	R/W	0x0
dpwrdn_en	[1]	**Dynamic Power Down** 0x0 = Disable, 0x1 = Enable	R/W	0x0
clk_stop_en	[0]	**Dynamic Clock Control** 0x0 = Always running, 0x1 = Stops during idle periods This feature is only supported with LPDDR2/LPDDR3. Refer to chapter 1.5.4 . Clock stop for detailed information.	R/W	0x1

1.11.4 Memory Chip0 Configuration Register (MemConfig0, R/W, Address Offset=0x0008)

Field	Bit	Description	R/W	Initial State
Reserved	[31:16]	Should be zero		0x0
chip_map	[15:12]	**Address Mapping Method (AXI to Memory)** 0x0 = Linear ({bank, row, column, width}), 0x1 = Interleaved ({row, bank, column, width}), 0x2 ~ 0xf = Reserved	R/W	0x1
chip_col	[11:8]	**Number of Column Address Bits** 0x0 = Reserved, 0x1 = Reserved, 0x2 = 9 bits, 0x3 = 10 bits, 0x4 = Reserved, 0x5 ~ 0xf = Reserved	R/W	0x3
chip_row	[7:4]	**Number of Row Address Bits** 0x0 = 12 bits, 0x1 = 13 bits, 0x2 = 14 bits, 0x3 = 15 bits, 0x4 = 16 bits, 0x5 ~ 0xf = Reserved	R/W	0x1
chip_bank	[3:0]	**Number of Banks** 0x0 = Reserved, 0x1 = Reserved, 0x2 = 4 banks, 0x3 = 8 banks, 0x4 ~ 0xf = Reserved	R/W	0x2

1.11.5 Memory Chip1 Configuration Register (MemConfig1, R/W, Address Offset=0x000C)

Field	Bit	Description	R/W	Initial State
Reserved	[31:16]	Should be zero		0x0
chip_map	[15:12]	**Address Mapping Method (AXI to Memory)** 0x0 = Linear ({bank, row, column, width}), 0x1 = Interleaved ({row, bank, column, width}), 0x2 ~ 0xf = Reserved	R/W	0x1
chip_col	[11:8]	**Number of Column Address Bits** 0x0 = Reserved, 0x1 = Reserved, 0x2 = 9 bits, 0x3 = 10 bits, 0x4 = Reserved, 0x5 ~ 0xf = Reserved	R/W	0x3
chip_row	[7:4]	**Number of Row Address Bits** 0x0 = 12 bits, 0x1 = 13 bits, 0x2 = 14 bits, 0x3 = 15 bits, 0x4 = 16 bits, 0x5 ~ 0xf = Reserved	R/W	0x1
chip_bank	[3:0]	**Number of Banks** 0x0 = Reserved, 0x1 = Reserved, 0x2 = 4 banks, 0x3 = 8 banks, 0x4 ~ 0xf = Reserved	R/W	0x2

1.11.6 Memory Direct Command Register (DirectCmd, R/W, Address Offset=0x0010)

Field	Bit	Description	R/W	Initial State
Reserved	[31:28]	Should be zero.		0x0
cmd_type	[27:24]	**Type of Direct Command** 0x0 = MRS/EMRS (mode register setting), 0x1 = PALL (all banks precharge), 0x2 = PRE (per bank precharge), 0x3 = DPD (deep power down), 0x4 = REFS (self refresh), 0x5 = REFA (auto refresh), 0x6 = CKEL (active/precharge power down), 0x7 = NOP (exit from active/precharge power down or deep power down, 0x8 = REFSX (exit from self refresh), 0x9 = MRR (mode register reading), 0xa = ZQINIT(ZQ calibration init.) 0xb = ZQOPER(ZQ calibration long) 0xc = ZQCS(ZQ calibration short) 0xd ~ 0xf = Reserved When a direct command is issued, AXI masters must not access memory. It is strongly recommended to check the command queue's state by ConControl.chip0/1_empty and the chip FSM in the ChipStatus register before issuing a direct command. The chip status must be checked before issuing a direct command. And clk_stop_en, dynamic power down, dynamic self refresh, force precharge function (MemControl register) and sl_dll_dyn_con (PhyControl0 register) must be disabled. MRS/EMRS or MRR commands should be issued if all banks are in idle state. If MRS/EMRS or MRR is issued to LPDDR2-S4/LPDDR3, the CA pins must be mapped as follows. MA[7:0] = {cmd_addr[1:0], cmd_bank[2:0], cmd_addr[12:10]}, OP[7:0] = cmd_addr[9:2] In DDR3, self refresh related timing such as tCKESR/tCKSRE/tCKSRX should be check by software. Note that do not write reserved value to this field.	R/W	0x0
Reserved	[23:21]	Should be zero.		0x0
cmd_chip	[20]	**Chip Number to send the direct command to** 0 = Chip 0 1 = Chip 1	R/W	0x0
Reserved	[19]	Should be zero.		0x0
cmd_bank	[18:16]	**Related Bank Address when issuing a direct command** To send a direct command to a chip, additional information such	R/W	0x0

		as the bank address is required. This register is used in such situations.		
cmd_addr	[15:0]	**Related Address value when issuing a direct command** To send a direct command to a chip, additional information such as the address is required. This register is used in such situations.	R/W	0x0

1.11.7 Precharge Policy Configuration Register (PrechConfig, R/W, Address Offset=0x14)

Field	Bit	Description	R/W	Initial State
tp_cnt	[31:24]	**Timeout Precharge Cycles** 0xn = n cclk cycles, The minimum vlaue of this field is 0x2 If the timeout precharge function (MemControl.tp_en) is enabled and the timeout precharge counter becomes zero, the controller forces the activated memory bank into the pre-charged state. Refer to chapter 1.6.2 .Timeout precharge for detailed information.	R/W	0xFF
Reserved	[23:16]	Should be zero		0x0
chip1_policy	[15:8]	**Memory Chip1 Precharge Bank Selective Policy** 0x0 = Open page policy, 0x1 = Close page (auto precharge) policy chip1_policy[n], n is the bank number of chip1. Open Page Policy: After a READ or WRITE, the row that was accessed is left open. Close Page (Auto Precharge) Policy: Right after a READ or WRITE command, memory devices automatically precharges the bank. This is a bank selective precharge policy. For example, if chip1_policy[2] is 0x0, bank2 of chip1 has an open page policy and if chip1_policy[6] is 0x1, bank6 of chip1 has a close page policy. Refer to chapter 1.6.1 . Bank Selective Precharge for detailed information.	R/W	0x0
chip0_policy	[7:0]	**Memory Chip0 Precharge Bank Selective Policy** 0x0 = Open page policy, 0x1 = Close page (auto precharge) policy Chip0_policy[n], n is the bank number of chip0. This is for memory chip0.	R/W	0x0

1.11.8 PHY Control0 Register (PhyControl0, R/W, Address Offset=0x0018)

Field	Bit	Description	R/W	Initial State
mem_term_en	[31]	**Termination Enable for Memory** At high-speed memory operation, it can be necessary to turn on termination resistance in memory devices. This register control s an ODT pin of a memory device.	R/W	0x0
phy_term_en	[30]	**Termination Enable for PHY** At high-speed memory operation, it can be necessary to turn on termination resistance in the PHY.	R/W	0x0
ctrl_shgate	[29]	**Duration of DQS Gating Signal** This field controls the gate control signal In LPDDR2-S4/LPDDRD3, this field should be 1'b0 regard-less of clock frequency. In DDR3, according to memory clock, set the value like be-low. 1'b0 = (gate signal length = "burst length / 2" (<= 200MHz)) 1'b1 = (gate signal length = "burst length / 2" – 1 (> 200MHz))		0x0
ctrl_pd	[28:24]	**Input Gate for Power Down** If this field is set, input buffer is off for power down. This field should be 0 for normal operation. Ctrl_pd[3:0] = for each data slice, Ctrl_pd[4] = for control slice.	R/W	0x0
Reserved	[23:7]	Should be zero		0x0
dqs_delay	[6:4]	**Delay Cycles for DQS Cleaning** This register is to enable PHY to clean incoming DQS sig-nals delayed by external circumstances. If DQS is coming with read latency plus n memory clock cycles, this registers must be set to n memory clock cycles.	R/W	0x0
fp_resync	[3]	**Force DLL Resyncronization**	R/W	0x0
Reserved	[2]	Should be zero		0x0
sl_dll_dyn_con	[1]	**Turn On PHY Slave DLL Dynamically** 0 = Disable 1 = Enable	R/W	0x0
mem_term_chips	[0]	Memory termination between chips 0 = Disable 1 = Enable This field is only valid when num_chip is 0x1(2 chips) in MemControl register and DDR3.	R/W	0x0

1.11.9 Dynamic Power Down Configuration Register (PwrdnConfig, R/W, Address Offset=0x0028)

Field	Bit	Description	R/W	Initial State
dsref_cyc	[31:16]	**Number of Cycles for dynamic self refresh entry** 0xn = n aclk cycles, The minimum value of this field is 0x2. If the command queue is empty for n+1 cycles, the controller forces memory devices into self refresh state. Refer to chapter 1.5.3 . Dynamic self refresh for detailed information.	R/W	0xFFFF
Reserved	[15:8]	Should be zero		0x0
dpwrdn_cyc	[7:0]	**Number of Cycles for dynamic power down entry** 0xn = n aclk cycles, The minimum value of this field is 0x2. If the command queue is empty for n+1 cycles, the controller forces the memory device into active/precharge power down state. Refer to chapter 1.5.2 . Dynamic power down for detailed information.	R/W	0xFF

1.11.10 AC Timing Register for Periodic ZQ(ZQCS) of Memory (TimingPZQ, R/W, Address Offset=0x002C)

Field	Bit	Description	R/W	Initial State
Reserved	[31:24]	Should be zero		0x0
t_pzq	[23:0]	Average Periodic ZQ Interval(Only in DDR3) tREFI(t_refi * T(rclk)) * t_pzq should be less than or equal to the minimum value of memory periodic ZQ interval, for example, if rclk frequency is 12MHz, t_refi is set to 93 and ZQ interval is 128ms then the following value should be programmed into it: 128 ms * 12 MHz / 93 = 16516 The minimum value is 2.		0x4084

1.11.11 AC Timing Register for Auto Refresh of Memory (TimingAref, R/W, Address Offset=0x0030)

Field	Bit	Description	R/W	Initial State
Reserved	[31:16]	Should be zero		0x0
t_refi	[15:0]	Average Periodic Refresh Interval t_refi * T(rclk) should be less than or equal to the minimum value of memory tREFI (all bank), for example, for the all bank refresh period of 7.8us, and an rclk frequency of 12MHz, the following value should be programmed into it: 7.8 us * 12 MHz = 93	R/W	0x5D

1.11.12 AC Timing Register for the Row of Memory (TimingRow, R/W, Address Offset=0x0034)

Field	Bit	Description	R/W	Initial State
t_rfc	[31:24]	Auto refresh to Active / Auto refresh command period, in cclk cycles t_rfc * T(cclk) should be greater than or equal to the minimum value of memory tRFC and the minimum value is 17 if PHY is running with dll on. In FPGA with low frequency and dll is off, the minimum value is 3.	R/W	0x1F
t_rrd	[23:20]	Active bank A to Active bank B delay, in cclk cycles t_rrd * T(cclk) should be greater than or equal to the minimum value of memory tRRD. The minimum value is 2.	R/W	0x2
t_rp	[19:16]	Precharge command period, in cclk cycles t_rp * T(cclk) should be greater than or equal to the minimum value of memory tRP. The minimum value is 2.	R/W	0x3
t_rcd	[15:12]	Active to Read or Write delay, in cclk cycles	R/W	0x3

		t_rcd * T(cclk) should be greater than or equal to the minimum value of memory tRCD + T(cclk)/2. For example, tRCD in memory specification is 13.75ns and cclk is 3.0ns, t_rcd * 3ns >= 13.75ns + 1.5ns The right value for t_rcd is 6. The minimum value is 2.		
t_rc	[11:6]	Active to Active period, in cclk cycles t_rc * T(cclk) should be greater than or equal to the minimum value of memory tRC. The minimum value is 2.	R/W	0xA
t_ras	[5:0]	Active to Precharge command period, in cclk cycles t_ras * T(cclk) should be greater than or equal to the minimum value of memory tRAS + T(cclk)/2. For example, tRAS in memory specification is 35ns and cclk is 3.0ns. t_ras * 3ns >= 35ns + 1.5ns The right value for t_ras is 13. The minimum value is 2.	R/W	0x6

1.11.13 AC Timing Register for the Data of Memory (TimingData, R/W, Address Offset=0x0038)

Field	Bit	Description	R/W	Initial State
t_wtr	[31:28]	Internal write to Read command delay, in cclk cycles t_wtr * T(cclk) should be greater than or equal to the minimum value of memory tWTR In LPDDR2-S4/LPDDR3 t_wtr is max(2tCK, tWTR) andIn DDR3 t_wtr is max(4tCK, tWTR). And then this value should be changed in cclk cycles. The minimum value is 2.	R/W	0x1
t_wr	[27:24]	Write recovery time, in cclk cycles t_wr * T(cclk) should be greater than or equal to the minimum value of memory tWR. The minimum value is 2.	R/W	0x2
t_rtp	[23:20]	Internal read to Precharge command delay, in cclk cycles t_rtp * T(cclk) should be greater than or equal to the minimum value of memory tRTP. The minimum value is 2.	R/W	0x3
Reserved	[19:17]	Should be zero	R/W	0x0
t_w2w_c2c	[16]	Additional Write to Write delay in chip to chip case in cclk cycles. If mem_term_en of PhyControl0 register(offset addr = 0x18) is enabled, then this value will be set to 1 automatically.	R/W	0x0
t_r2r_c2c	[15]	Additional Read to Read delay in chip to chip case in cclk cycles. If mem_term_chips of PhyControl0 register(offset addr = 0x18) is enabled, then this value will be set to 1 automatically.	R/W	0x0
dqsck	[14:12]	tDQSCK in memory clock cycles In DDR3, this field should set to 0. In LPDDR2/3, this field should be set to RU(tDQSCK max/tCK). tDQSCK max is 5.5ns in LPDDR2/3.		0x3
wl	[11:8]	Write data latency (for LPDDR2-S4/LPDDR3/DDR3), in memory clock cycles wl should be greater than or equal to the minimum value of memory WL. There is no restriction with LPDDR2-S4 but there is restriction with LPDDR3 and DDR3 like below. In LPDDR3, the minimum wl is 5 and in DDR3, the minimum wl is 6.	R/W	0x6
Reserved	[7:4]	Should be zero		0x0
rl	[3:0]	Read data latency (for LPDDR2-S4/LPDDR3/DDR3), in memory clock cycles rl should be greater than or equal to the minimum value of memory RL	R/W	0xC

NOTE: tDAL (Auto precharge write recovery + precharge time) = t_wr + t_rp (automatically calculated)

1.11.14 AC Timing Register for the Power modes of Memory (TimingPower, R/W, Address Offset=0x003C)

Field	Bit	Description	R/W	Initial State
t_faw	[31:26]	Four Active Window(for LPDDR2-S4/LPDDR3/DDR3) t_faw * T(cclk) should be greater than or equal to the minimum value of memory tFAW. The minimum value is 2.	R/W	0xE
t_xsr	[25:16]	Self refresh exit power down to next valid command delay, in cycles t_xsr * T(cclk) should be greater than or equal to the minimum value of memory tXSR. In DDR3, this value should be set to tXSDLL. The minimum value is 2.	R/W	0x1B
t_xp	[15:8]	Exit power down to next valid command delay, in cycles t_xp * T(cclk) should be greater than or equal to the minimum value of memory tXP In DDR3, this value should set to tXPDLL. In DDR3 even though “fast exit” is programmed in MRS, tXPDLL is applied. In DDR3, tXPDLL is likely max(10nCK,24ns). So note that t_xp should set to max(5nCCLK,24ns/CCLK period). The minimum value is 2.	R/W	0x4
t_cke	[7:4]	CKE minimum pulse width (minimum power down mode duration), in cycles t_cke should be greater than or equal to the minimum value of memory tCKE. The minimum value is 2.	R/W	0x2
t_mrd	[3:0]	Mode Register Set command period, in cycles t_mrd should be greater than or equal to the minimum value of memory tMRD In DDR3, this parameter should be set to tMOD value. The minimum value is 2.	R/W	0x2

1.11.15 PHY Status Register (PhyStatus, Read Only, Address Offset=0x0040)

Field	Bit	Description	R/W	Initial State
Reserved	[31:15]	Should be zero		0x0
rdlvl_complete	[14]	Read Level Completion	R	0x0
Reserved	[13:4]	Should be zero		0x0
dfi_init_complete	[3]	DFI PHY initialization complete 0 = Initialization has not been finished 1 = Initialization has been finished		0x0
Reserved	[2:0]	Should be zero		0x0

1.11.16 Memory ChipStatus Register (ChipStatus, Read Only, Address Offset=0x0048)

Field	Bit	Description	R/W	Initial State
Reserved	[31:16]	Should be zero		0x0
chip_dpd_state	[15:12]	Chip is in the deep powerdown state	R	0x0
chip_sref_state	[11:8]	Chip is in the self-refresh state	R	0x0
chip_pd_state	[7:4]	Chip is in the powerdown state	R	0x0
chip_busy_state	[3:0]	Chip is in the busy state [0] = chip0 busy state [1] = chip1 busy state [2] = chip2 busy state [3] = chip3 busy state	R	0x0

1.11.17 Memory Mode Registers Status Register (MrStatus, Read Only, Address Offset=0x0054)

Field	Bit	Description	R/W	Initial State
Reserved	[31:8]	Should be zero		0x0
mr_status	[7:0]	Mode Registers Status	R	0x0

1.11.18 Quality of Service Control Register n (QosControl n, R/W, Address Offset=0x0060 + 8n (n=0~15, integer))

Field	Bit	Description	R/W	Initial State
Reserved	[31:12]	Should be zero		0x0
cfg_qos	[11:0]	QoS Cycles 0xn = n aclk cycles The matched ARID uses this value for its timeout counters instead of ConControl.timeout_cnt.	R/W	0xFFF

1.11.19 Write Training Configuration Register (WrTraConfig, R/W, Address Offset=0x00F4)

Field	Bit	Description	R/W	Initial State
row_addr	[31:16]	Row Address for Write Training	R/W	0x0
Reserved	[15:4]	Should be zero		0x0
bank	[3:1]	Bank Address for Write Training	R/W	0x0
write_training_en	[0]	Write Training Enable Use this field to issue ACT command. Before setting this field, below things should be finished. Please see PHY manual. - Set write latency before write training(PHY's control register 26) - Set write training mode(PHY's control register 2) 0x0 = Disable, 0x1 = Enable (Issue ACT command)	R/W	0x0

1.11.20 Read Leveling Configuration Register (RdlvlConfig, R/W, Address Offset=0x00F8)

Field	Bit	Description	R/W	Initial State
Reserved	[31:2]	Should be zero		0x0
ctrl_rdlvl_data_en	[1]	Data eye training enable	R/W	0x0
ctrl_rdlvl_gate_en	[0]	Gate training enable This is only valid for DDR3 case. If LPDDR2-S4/LPDDR3 is used, this field must be set to 0x0. When ctrl_rdlvl_en = 1, Read leveling offset vlaues will be used instead of ctrl_offsetr*. If read leveling is used, this vlaue should be high during operation. This field should be set after dfi_init_complete is asserted. 0x0 = Disable, 0x1 = Enable	R/W	0x0

1.11.21 BRB Reservation Control Register (BRBRSVCONTROL, R/W, Address Offset=0x0100)

Field	Bit	Description	R/W	Initial State
Reserved	[31:8]	Should be zero		0x0
brb_rsv_en_w3	[7]	Enable Write-BRB reservation for AXI port 3	R/W	0x0
brb_rsv_en_w2	[6]	Enable Write-BRB reservation for AXI port 2	R/W	0x0
brb_rsv_en_w1	[5]	Enable Write-BRB reservation for AXI port 1	R/W	0x0
brb_rsv_en_w0	[4]	Enable Write-BRB reservation for AXI port 0	R/W	0x0
brb_rsv_en_r3	[3]	Enable Read-BRB reservation for AXI port 3	R/W	0x0
brb_rsv_en_r2	[2]	Enable Read-BRB reservation for AXI port 2	R/W	0x0
brb_rsv_en_r1	[1]	Enable Read-BRB reservation for AXI port 1	R/W	0x0
brb_rsv_en_r0	[0]	Enable Read-BRB reservation for AXI port 0	R/W	0x0

1.11.22 BRB Reservation Configuation Register (BRBRSVCONFIG, R/W, Address Offset=0x0104)

Field	Bit	Description	R/W	Initial State
brb_rsv_th_w3	[31:28]	Write-BRB reservation threshold for AXI port 3 Write request from AXI port3 does not serviced when Write-BRB reservation is enabled for AXI port 3 and the occupancy of corresponding Write-BRB exceeds brb_rsv_th_w3.	R/W	0x8
brb_rsv_th_w2	[27:24]	Enable Write-BRB reservation for AXI port 2 Write request from AXI port2 does not serviced when Write-BRB reservation is enabled for AXI port 2 and the occupancy of corresponding Write-BRB exceeds brb_rsv_th_w2.	R/W	0x8
brb_rsv_th_w1	[23:20]	Enable Write-BRB reservation for AXI port 1 Write request from AXI port1 does not serviced when Write-BRB reservation is enabled for AXI port 1 and the occupancy of corresponding Write-BRB exceeds brb_rsv_th_w1.	R/W	0x8
brb_rsv_th_w0	[19:16]	Enable Write-BRB reservation for AXI port 0 Write request from AXI port0 does not serviced when Write-BRB reservation is enabled for AXI port 0 and the occupancy of corresponding Write-BRB exceeds brb_rsv_th_w0.	R/W	0x8
brb_rsv_th_r3	[15:12]	Enable Read-BRB reservation for AXI port 3 Read request from AXI port3 does not serviced when Read-BRB reservation is enabled for AXI port 3 and the occupancy of corresponding Read-BRB exceeds brb_rsv_th_w3.	R/W	0x8
brb_rsv_th_r2	[11:8]	Enable Read-BRB reservation for AXI port 2 Read request from AXI port2 does not serviced when Read-BRB reservation is enabled for AXI port 2 and the occupancy of corresponding Read-BRB exceeds brb_rsv_th_w2.	R/W	0x8
brb_rsv_th_r1	[7:4]	Enable Read-BRB reservation for AXI port 1 Read request from AXI port1 does not serviced when Read-BRB reservation is enabled for AXI port 1 and the occupancy of corresponding Read-BRB exceeds brb_rsv_th_w1.	R/W	0x8
brb_rsv_th_r0	[3:0]	Enable Read-BRB reservation for AXI port 0 Read request from AXI port0 does not serviced when Read-BRB reservation is enabled for AXI port 0 and the occupancy of corresponding Read-BRB exceeds brb_rsv_th_w0.	R/W	0x8

1.11.23 BRB QoS Configuation Register (BRBQOSCONFIG, R/W, Address Offset=0x0108)

Field	Bit	Description	R/W	Initial State
Reserved	[31:12]	Should be zero		0x0
brb_qos_timer_dec	[11:0]	BRB timer decrementing size for QoS. The timer for request in BRB decreases by brb_qos_timer_dec for the following cases. - When the BRB is full - When the request is from the AXI port whose data buffer is full.	R/W	0x10

1.11.24 Memory Chip0 Base Configuration Register (MemBaseConfig0, R/W, Address Offset=0x010C)

Field	Bit	Description	R/W	Initial State
Reserved	[31:27]	Should be zero		0x0
chip_base	[26:16]	**AXI Base Address** AXI base address [34:24] = chip_base, For example, if chip_base = 0x20, then AXI base address of memory chip0 becomes 0x2000_0000.	R/W	0x20
Reserved	[15:11]	Should be zero		0x0
chip_mask	[10:0]	**AXI Base Address Mask** Upper address bit mask to determine AXI offset address of memory chip0. 0 = Corresponding address bit is not to be used for comparison 1 = Corresponding address bit is to be used for comparison For example, if chip_mask = 0x7F8, then AXI offset address becomes 0x0000_0000 ~ 0x07FF_FFFF. If AXI base address of memory chip0 is 0x2000_0000, then memory chip0 has an address range of 0x2000_0000 ~ 0x27FF_FFFF.	R/W	0x7F8

1.11.25 Memory Chip1 Base Configuration Register (MemBaseConfig1, R/W, Address Offset=0x0110)

Field	Bit	Description	R/W	Initial State
Reserved	[31:27]	Should be zero		0x0
chip_base	[26:16]	**AXI Base Address** AXI base address [34:24] = chip_base, For example, if chip_base = 0x28, then AXI base address of chip1 becomes 0x2800_0000.	R/W	0x28
Reserved	[15:11]	Should be zero		0x0
chip_mask	[10:0]	**AXI Base Address Mask** Upper address bit mask to determine AXI offset address of memory chip1. 0 = Corresponding address bit is not to be used for comparison 1 = Corresponding address bit is to be used for comparison For example, if chip_mask = 0x7F0, then AXI offset address becomes 0x0000_0000 ~ 0x0FFF_FFFF. If AXI base address of memory chip1 is 0x2800_0000, then memory chip1 has an address range of 0x2800_0000 ~ 0x37FF_FFFF.	R/W	0x7F8

1.11.26 Write Leveling Configuration Register0 (WRLVLCONFIG0, R/W, Address Offset=0x0120)

Field	Bit	Description	R/W	Initial State
Reserved	[31:8]	Should be zero		0x0
t_wlo	[7:4]	**Write Leveling Ouput Delay** t_wlo * T(PCLK) should be greater than or equal to the minimum value of memory tWLO and the minimum value is 1.	R/W	0x1
Reserved	[3:1]	Should be zero		0x0
odt_on	[0]	**Turn On ODT for Write Leveling** 0x0 = ODT Turn off 0x1 = ODT Turn on, This field is only for write leveling. Turn on before write leveling and turn off after write leveling is finished. Write leveling procedure is MRS for Write leveling - ODT on - wrlvl_wrdata_en - Read ctrl_io_rdata - Change SDLL code of PHY. Refer to PHY manual for details.	R/W	0x0

1.11.27 Write Leveling Configuration Register1 (WRLVLCONFIG1, R/W, Address Offset=0x0124)

Field	Bit	Description	R/W	Initial State
Reserved	[31:1]	Should be zero		0x0
wrlvl_wrdata_en	[0]	**Generate dfi_wrdata_en_p0 for Write Leveling** Generate 1cycle pulse of dfi_wrdata_en_p0 for write leveling. Write leveling is supported in DDR3 and LPDDR3. Note that if S/W write this field to 1 then, 1 cycle pulse of dfi_wrdata_en_p0 would be generated. Refer to PHY manual for write leveling.	R/W	0x0

1.11.28 Write Leveling Status Register (WRLVLSTATUS, R/W, Address Offset=0x0128)

Field	Bit	Description	R/W	Initial State
Reserved	[31:5]	Should be zero		0x0
wrlvl_fsm	[4:0]	**Write Leveling Status** 5'b0_0001: FSM IDLE 5'b0_0010: FSM_SETUP 5'b0_0100: FSM_ACCESS 5'b0_1000: FSM_DONE 5'b1_0000: FSM_TWLO wrlvl_eq is valid only when wrlvl_fsm is FSM_IDLE. Refer to PHY manual for write leveling.	R	0x0

		Performance events list numbers: {0x3,0x2,0x1,0x0} = {reserved, read, reserved, write} for port0 {0x7,0x6,0x5,0x4} = {reserved, read, reserved, write} for port1 {0xb,0xa,0x9,0x8} = {reserved, read, reserved, write} for port2 {0xf,0xe,0xd,0xc} = {reserved, read, reserved, write} for port3 {0x13,0x12,0x11,0x10} = read to bank[3:0] for port0 {0x17,0x16,0x15,0x14} = read to bank[7:4] for port0 {0x1b,0x1a,0x19,0x18} = write to bank[3:0] for port0 {0x1f,0x1e,0x1d,0x1c} = write to bank[7:4] for port0 {0x23,0x22,0x21,0x20} = read to bank[3:0] for port1 {0x27,0x26,0x25,0x24} = read to bank[7:4] for port1 {0x2b,0x2a,0x29,0x28} = write to bank[3:0] for port1 {0x2f,0x2e,0x2d,0x2c} = write to bank[7:4] for port1 {0x33,0x32,0x31,0x30} = read to bank[3:0] for port2 {0x37,0x36,0x35,0x34} = read to bank[7:4] for port2 {0x3b,0x3a,0x39,0x38} = write to bank[3:0] for port2 {0x3f,0x3e,0x3d,0x3c} = write to bank[7:4] for port2 {0x43,0x42,0x41,0x40} = read to bank[3:0] for port3 {0x47,0x46,0x45,0x44} = read to bank[7:4] for port3 {0x4b,0x4a,0x49,0x48} = write to bank[3:0] for port3 {0x4f,0x4e,0x4d,0x4c} = write to bank[7:4] for port3 {0x67 .. 0x50} = {reserved .. reserved} {0x6b,0x6a,0x69,0x68} = {reserved, cas scheduled on channel0, reserved, page miss on channel0} {0x6f,0x6e,0x6d,0x6c} = {reserved, reserved, read transfer on channel0, write transfer on channel0}		

1.11.29 Performance Event Configuration0 Register (PerevConfig0, R/W, Address Offset=0x0134)

Field	Bit	Description	R/W	Initial State
Reserved	[31:7]	Should be zero		0x0
perev0_sel	[6:0]	Performance event0 selection See above performance event number	R/W	0x0

1.11.30 Performance Event Configuration1 Register (PerevConfig1, R/W, Address Offset=0x0138)

Field	Bit	Description	R/W	Initial State
Reserved	[31:7]	Should be zero		0x0
perev1_sel	[6:0]	Performance event1 selection See above performance event number	R/W	0x0

1.11.31 Performance Event Configuration2 Register (PerevConfig2, R/W, Address Offset=0x013C)

Field	Bit	Description	R/W	Initial State
Reserved	[31:7]	Should be zero		0x0
perev2_sel	[6:0]	Performance event2 selection See above performance event number	R/W	0x0

1.11.32 Performance Event3 Configuration Register (PerevConfig3, R/W, Address Offset=0x0140)

Field	Bit	Description	R/W	Initial State
Reserved	[31:7]	Should be zero		0x0
perev3_sel	[6:0]	Performance event3 selection See above performance event number	R/W	0x0

1.11.33 CTRL_IO_RDATA Register (CTRL_IO_RDATA, R, Address Offset=0x0150)

Field	Bit	Description	R/W	Initial State
ctrl_io_rdata	[31:0]	ctrl_io_rdata from PHY	R	0x0

1.11.34 CA Calibration Configuration Register0 (CACAL_CONFIG0, R/W, Address Offset=0x0160)

Field	Bit	Description	R/W	Initial State
dfi_address_p0	[31:12]	**dfi_address_p0 value for CA Calibration** This value would be the expected value for comparing the address pattern received from memory.		0x3FF
Reserved	[11:8]	Should be zero		0x0
t_adr	[7:4]	**CSN Low to Data Output Delay** t_adr * T(PCLK) should be greater than or equal to the minimum value of memory tADR and the minimum value is 1.	R/W	0x1
Reserved	[3:1]	Should be zero		0x0
deassert_cke	[0]	**Deassert CKE for CA Calibration** 0x0 = Put CKE pin to normal operation 0x1 = Put CKE pin to low This field is only for CA calibration. Deassert CKE before CA calibration and put to normal operation after CA calibration is finished. CA calibration procedure is MRS for CA calibration - Deassert CKE - cacal_csn - Read ctrl_io_rdata - Change SDLL code of PHY. Refer to PHY manual for details.	R/W	0x0

1.11.35 CA Calibration Configuration Register (CACAL_CONFIG1, R/W, Address Offset=0x0164)

Field	Bit	Description	R/W	Initial State
Reserved	[31:1]	Should be zero		0x0
cacal_csn	[0]	**Generate dfi_csn_p0 for CA Calibration** Generate 1cycle pulse of dfi_csn_p0 for CA calibration. CA calibration is supported in LPDDR3. Note that if S/W writes this field to 1 then, 1 cycle pulse of dfi_csn_p0 would be generated. Refer to PHY manual for CA calibration.	R/W	0x0

1.11.36 CA Calibration Status Register (CACAL_STATUS, R, Address Offset=0x0168)

Field	Bit	Description	R/W	Initial State
Reserved	[31:5]	Should be zero		0x0
cacal_fsm	[4:0]	**Write Leveling Status** 5'b0_0001: FSM IDLE 5'b0_0010: FSM_SETUP 5'b0_0100: FSM_ACCESS 5'b0_1000: FSM_DONE 5'b1_0000: FSM_TADR CTRL_IO_RDATA are valid only when wrlvl_fsm is FSM_IDLE. Refer to PHY manual for CA calibration.	R	0x0

1.11.37 Performance Monitor Control Register (PMNC_PPC, R/W, Address Offset=0xE000)

Field	Bit	Description	R/W	Initial State
Reserved	[31:18]	Should be zero		0x0
START MODE	[16]	PPC Start Mode 0x0 = SW(by CPU) 0x1 = HW(by SYSCON)	R/W	0x0
Reserved	[15:4]	Should be zero		0x0
CC DIVIDER	[3]	Cycle count divider 0x0 = counts every processor clock cycle, reset value 0x1 = counts every 64th processor clock cycle	R/W	0x0
CC RESET	[2]	Cycle counter reset 0x0 = no action 0x1 = resets cycle counter, CCNT, to zero	W	0x0
PPC COUNTER RESET	[1]	Performance counter reset 0x0 = no action 0x1 = resets all performance counters to zero	W	0x0

PPC ENABLE	[0]	Enable bit 0x0 = disables all counters including CCNT 0x1 = enables all counters including CCNT When you read it, 1 means it's counting and 0 means it's idle (stop counting). You can write it only when the start mode is set to be 0 (PPMU is started by CPU). At this time, you can write this bit by 1 to start counting and write it by 0 to stop the counting. When the start mode is set to be 1 (PPMU is started by SYSCON), you only can read it and get the status of the PPMU. At this time, PPMU is controlled by external trigger. When external trigger is 1, counting starts, and when external trigger is 0, counting stops.	R/W	0x0

1.11.38 Count Enable Set Register (CNTENS_PPC, R/W, Address Offset=0xE010)

Field	Bit	Description	R/W	Initial State
CCNT	[31]	Enable cycle counter	R/W	0x0
Reserved	[30:4]	Should be zero		0x0
PMCNT3	[3]	Enable Counter 3	R/W	0x0
PMCNT2	[2]	Enable Counter 2	R/W	0x0
PMCNT1	[1]	Enable Counter 1	R/W	0x0
PMCNT0	[0]	Enable Counter 0	R/W	0x0

1.11.39 Count Enable Clear Register (CNTENC_PPC, R/W, Address Offset=0xE020)

Field	Bit	Description	R/W	Initial State
CCNT	[31]	Disable cycle counter	R/W	0x0
Reserved	[30:4]	Should be zero		0x0
PMCNT3	[3]	Disable Counter 3	R/W	0x0
PMCNT2	[2]	Disable Counter 2	R/W	0x0
PMCNT1	[1]	Disable Counter 1	R/W	0x0
PMCNT0	[0]	Disable Counter 0	R/W	0x0

1.11.40 Interrupt Enable Set Register (INTENS_PPC, R/W, Address Offset=0xE030)

Field	Bit	Description	R/W	Initial State
CCNT	[31]	CCNT overflow interrupt enable	R/W	0x0
Reserved	[30:4]	Should be zero		0x0
PMCNT3	[3]	Counter 3 overflow interrupt enable	R/W	0x0
PMCNT2	[2]	Counter 2 overflow interrupt enable	R/W	0x0
PMCNT1	[1]	Counter 1 overflow interrupt enable	R/W	0x0
PMCNT0	[0]	Counter 0 overflow interrupt enable	R/W	0x0

1.11.41 Interrupt Enable Clear Register (INTENC_PPC, R/W, Address Offset=0xE040)

Field	Bit	Description	R/W	Initial State
CCNT	[31]	CCNT overflow interrupt disable	R/W	0x0
Reserved	[30:4]	Should be zero		0x0
PMCNT3	[3]	Counter 3 overflow interrupt disable	R/W	0x0
PMCNT2	[2]	Counter 2 overflow interrupt disable	R/W	0x0
PMCNT1	[1]	Counter 1 overflow interrupt disable	R/W	0x0
PMCNT0	[0]	Counter 0 overflow interrupt disable	R/W	0x0

1.11.42 Overflow Flag Status Register (FLAG_PPC, R/W, Address Offset=0xE050)

Field	Bit	Description	R/W	Initial State
CCNT	[31]	Cycle counter overflow flag	R/W	0x0
Reserved	[30:4]	Should be zero		0x0
PMCNT3	[3]	Counter 3 overflow flag	R/W	0x0
PMCNT2	[2]	Counter 2 overflow flag	R/W	0x0
PMCNT1	[1]	Counter 1 overflow flag	R/W	0x0
PMCNT0	[0]	Counter 0 overflow flag	R/W	0x0

1.11.43 Cycle Count Register (CCNT_PPC, R/W, Address Offset=0xE100)

Field	Bit	Description	R/W	Initial State
CCNT	[31:0]	CCNT Register contain an event count	R/W	0x0

1.11.44 Performance Monitor Count0 Register (PMCNT0_PPC, R/W, Address Offset=0xE110)

Field	Bit	Description	R/W	Initial State
PMCNT	[31:0]	PMCNT Register contain an event count	R/W	0x0

1.11.45 Performance Monitor Count1 Register (PMCNT1_PPC, R/W, Address Offset=0xE120)

Field	Bit	Description	R/W	Initial State
PMCNT	[31:0]	PMCNT Register contain an event count	R/W	0x0

1.11.46 Performance Monitor Count2 Register (PMCNT2_PPC, R/W, Address Offset=0xE130)

Field	Bit	Description	R/W	Initial State
PMCNT	[31:0]	PMCNT Register contain an event count	R/W	0x0

1.11.47 Performance Monitor Count3 Register (PMCNT3_PPC, R/W, Address Offset=0xE140)

Field	Bit	Description	R/W	Initial State
PMCNT	[31:0]	PMCNT Register contain an event count	R/W	0x0