Hardware System Verification (HSV)
Vertical Solutions Engineering (VSE)

LPDDR5
Palladium/Protium Memory Model
User Guide

Document Version: **1.4**

Document Date: **April 2020**

Contents

List of Figures

List of Tables

General Information

The Cadence Memory Model Portfolio provides memory device models for the Cadence Palladium/Protium series systems. Optimizing the acceleration and/or emulation flow on these platforms for MMP memory models may require information outside the scope of the MMP user guides and related MMP documentation.

1.1 Related Publications

For basic information regarding emulation and acceleration, please refer to the following documents:

For Palladium:

VXE User Guide
VXE Library Developer's Guide
VXE Known Problems and Solutions
VXE Command Reference Manual
Palladium Z1 Planning and Installation Guide
Palladium Target System Developer's Guide
What's New in VXE

For Protium:

Protium User Guide
Protium Command Guide
Protium Express Toolkit Command Reference Guide
Protium S1 Hardware Installation and Reference Guide
Protium X1 Hardware Installation and Reference Guide
Protium Release Information
Protium Known Problems and Solutions
What's New in Protium

LPDDR5 Palladium/Protium Memory Model

1. Introduction

The Cadence Palladium/Protium LPDDR5 model is based on the specification or datasheet listed below.
The model supports S16 and S32 types.

Table 1: Memory Model Standard or Datasheet Revision Info

Association or Vendor	Part	Reference Datasheet	Rev.	Revision Date
JEDEC	Jedec_lpddr5_*	February 2019 JEDEC LPDDR5 Specification JESD209-5.pdf	RevJESD209-5	February 2019
JEDEC	Jedec_lpddr5_x8_*	February 2019 JEDEC LPDDR5 Specification JESD209-5.pdf	RevJESD209-5	February 2019

2. Model Release Levels

All models in the Memory Model Portfolio are graded with a release level. This release level informs users of the current maturity and status of the model. All families in the library are graded at one of these levels.

The different levels give an overall indication of the amount of testing, level of quality and feature availability in the model. For details on supported features check the User Guide for that particular model family.

There are three release levels for models in the MMP release.

Release Level		Model Status	Available in Release	Listed in Catalog	Requires Beta Agreement
Mainstream Release	MR	Fully released and available in the catalog for all customers to use.	Yes	Yes	No
Emerging Release	ER	Model has successfully completed Beta engagement(s). Most, but not all features have been tested. Documentation is available.	No	Yes	Yes
Initial Release	IR	Model has completed initial development and has been released to Beta customer(s). The model may have missing features, may not be fully tested and may not have documentation. Model may contain defects.	No	Yes	Yes

Access to Initial Release and Emerging Release versions of the models will require a Beta Agreement to be signed before the model can be delivered.

3. Features

The table below lists which features are supported and which are unsupported.

Table 2: Features List of LPDDR5 Model

FEATURE	SUPPORT	NOTE
Initialization command sequence	Yes	After initialization sequence, init_done signal will be asserted 'HIGH'.
Power Down Enter/Exit	Yes	
CAS command with WCK2CK-sync	Yes	
Activate command	Yes	
Precharge command	Yes	
Burst Read command (READ, READ32)	Yes	
Burst Write command (WRITE, WRITE32)	Yes	
Masked Burst Write command	Yes	
Read/Write DBI function	Yes	
Burst Length 16, 32 and BL on the fly function	Yes	
All Read/Write latencies	Yes	
Mode Register Read/Write	Yes	
FSP-WR and FSP-OP features	Yes	
RDQS Pre-amble and Post-amble features	Yes	
Refresh/self-refresh/DSM command	Yes	REF, DSM, SRE and SRX commands are accepted for LPDDR5 model, but have no effect on the core memory. All bank refresh is supported.
Command Bus Training feature	Yes	
Read/Write FIFO command	Yes	
Read DQ Calibration command	Yes	
Write Leveling feature	Yes	
MPC command and its sub commands	Yes	
Data Copy	Yes	
Write X	Yes	
RDQS Interval Oscillator	Yes	
DVFS	Yes	
Post Package Repair	No	
ZQCal Reset function	No	
ODT feature	No	
PASR Bank and PASR Segment functions	No	
LPDDR5 x8 (Byte Mode)	Yes	
Timing parameter TWCK2DQI/TWCK2DQO	Yes	Supported in unit of half tWCK cycle
Link ECC feature	YES	Optional feature
DFE(Decision Feedback Equalization)	No	Optional feature
TRR(Target Row Refresh	No	Optional feature
Deep Sleep Mode	No	Optional feature
Enhanced WCK Always On Mode	No	Optional feature
Optimized Refresh Mode	No	Optional feature

FEATURE	SUPPORT	NOTE
DMI Output Behavior Mode	No	Optional feature
Unified NT-ODT Behavior	No	Optional feature

4. Verilog Macro Defines

The following table lists the optional Verilog macros which users may consider.

Table 3: LPDDR5 Optional Verilog Defines

`define Verilog Macro Purpose	Optional Verilog `define Values
RDQS Interval Oscillator feature support; define this optional Verilog macro to enable RDQS Interval Oscillator. MR35, MR36, MR37 and MR38 will report the DRAM RDQS Interval Oscillator count only when this macro is defined.	MMP_RDQSOSC
Support {row, bank, col} addressing mode.	MMP_RBC

5. Configurations

The following table lists the possible configurations. Not all configurations are available from all vendors. Please consult the appropriate vendor site for the detailed information on the parts they offer.

Table 4: Model Configurations

Data Width	Memory Density	2Gb		3Gb		4Gb		6Gb		8Gb		12Gb		16Gb		24Gb		32Gb	
	Type	S16	S32	S16	S32	S16	S32	S16	S32	S16	S32	S16	S32	S16	S32	S16	S32	S16	S32
	Bank Groups	4	0	4	0	4	0	4	0	4	0	4	0	4	0	4	0	4	0
	Banks/per Bank Group	4	8	4	8	4	8	4	8	4	8	4	8	4	8	4	8	4	8
x16	Row Address	R[12:0]		R[13:0] (R12=0 when R13=1)		R[13:0]		R[14:0] (R13=0 when R14=1)		R[14:0]		R[15:0] (R14=0 when R15=1)		R[15:0]		R[16:0] (R15=0 when R16=1)		R[16:0]	
	Column Address	C[5:0]		C[5:0]		C[5:0]		C[5:0]		C[5:0]		C[5:0]		C[5:0]		C[5:0]		C[5:0]	
	Burst Length	16	32	16	32	16	32	16	32	16	32	16	32	16	32	16	32	16	32
X8	Row Address	R[13:0]		R[14:0] (R13=0 when R14=1)		R[14:0]		R[15:0] (R14=0 when R15=1)		R[15:0]		R[16:0] (R15=0 when R16=1)		R[16:0]		R[17:0] (R16=0 when R17=1)		R[17:0]	
	Column Address	C[5:0]		C[5:0]		C[5:0]		C[5:0]		C[5:0]		C[5:0]		C[5:0]		C[5:0]		C[5:0]	
	Burst Length	16	32	16	32	16	32	16	32	16	32	16	32	16	32	16	32	16	32

6. Model Block Diagram

For the LPDDR5 x16 model, the width of CA bus is 7 bits; DMI is 2 bits; and DQ is 16 bits. WCK_t, WCK_c, RDQS_t and RDQS_c are 2-bit buses.

WCK2DQI_OSC_CLK and WCK2DQO_OSC_CLK are two optional clock inputs used for supporting the RDQS Interval Oscillator feature (see Table 3: LPDDR5 Optional Verilog Defines).

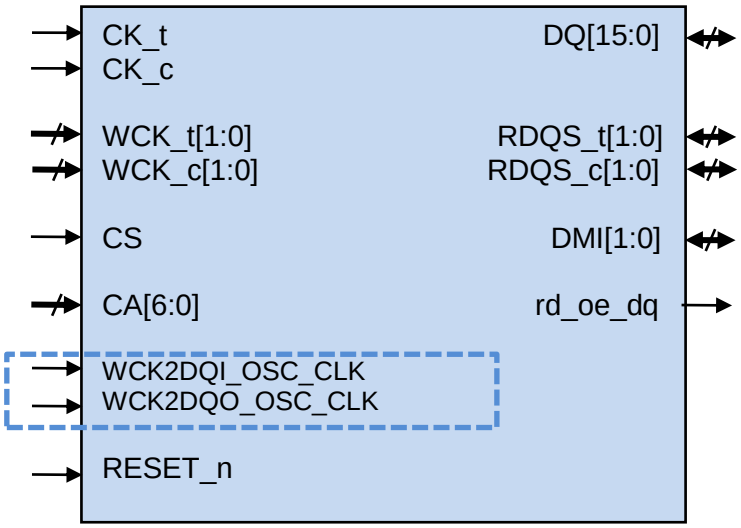


Figure 1: LPDDR5 x16 Model Port Diagram

For the LPDDR5 x8 model (also referred to as Byte Mode), the width of CA bus is 7 bits; DMI is 1 bit; and DQ is 8 bits. WCK_t, WCK_c RDQS_t and RDQS_c are each 1-bit.

WCK2DQI_OSC_CLK and WCK2DQO_OSC_CLK are two optional clock inputs used for supporting the RDQS Interval Oscillator feature.

The rd_oe_dq is an output port for LPDDR5 DQ output enable, which is only used for MMP DFI PHY model.

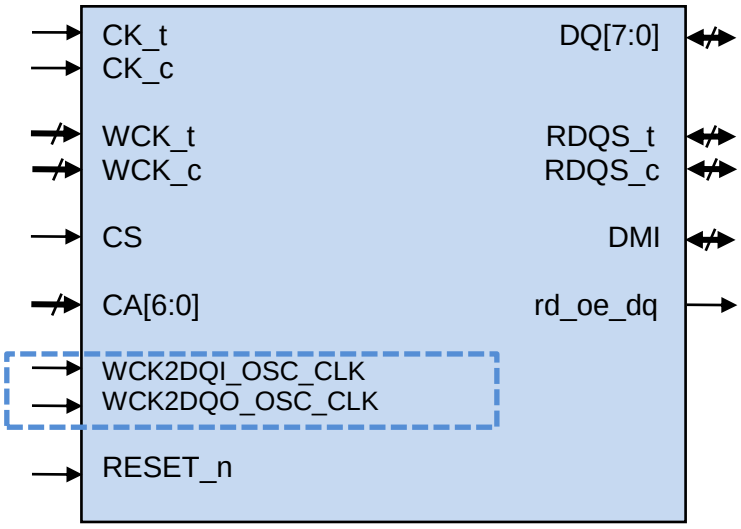


Figure 2: LPDDR5 x8 Model (Byte Mode) Port Diagram

7. I/O Signal Description

The table below lists and describes the model I/O signals.*

Table 5: Model I/O Signals

NAME	TYPE	DESCRIPTION
CK_t CK_c	Input	Clock: CK_t and CK_c are differential clock inputs. All address, command, and control input signals are sampled on the crossing of the positive edge of CK_t and the negative edge of CK_c. AC timings for CA parameters are referenced to CK.)
CS	Input	Chip Select: CS is part of the command code.
CA[6:0]	Input	Command/Address Inputs: CA signals provide the Command and Address inputs.
DQ[15:0]	I/O	Data Input/Output: Bi-directional data bus.
WCK_t[1:0] WCK_c[1:0]	Input	Write Clocks: WCK_t and WCK_c are differential clocks used for WRITE data capture and READ data output.
DMI[1:0]	I/O	Data Mask Inversion: DMI is a bi-directional signal which is driven HIGH when the data on the data bus is inverted, or driven LOW when the data is in its normal state. Data Inversion can be disabled via a mode register setting. Each byte of data has a DMI signal. This signal is also used along with the DQ signals to provide write data masking information to the DRAM. The DMI pin function—Data Inversion or Data mask—depends on Mode Register setting.
ZQ	Reference	*Not supported*
VDDQ, VDD1, VDD2H VDD2L	Supply	*Not supported*
VSS, VSSQ	GND	*Not supported*
RESET_n	Input	RESET: When asserted LOW, the RESET_n signal resets the die.

* JEDEC, “LPDDR5 Spec”, rev JESD209-5, February 2019.

NAME	TYPE	DESCRIPTION
RDQS_t[1:0]	I/O	Read Data Strobe: LPDDR5 can support read data strobe with RDQS_t pad in RDQS mode. RDQS_t can be supported with RDQS_c for differential RDQS mode option. And RDQS can be supported for single-ended RDQS mode option. RDQS_t[0] is associated with DQ[7:0], RDQS_c[1]is associated with DQ[15:8]. The functionality of RDQS_t pin depends on the MR20 OP[1:0] setting. RDQS_t is also used as a Parity pin at Write with Link Protection enabled.
RDQS_c[1:0]	I/O	Read Data Strobe: LPDDR5 can support read data strobe with RDQS_c pad in RDQS mode. RDQS_c can be supported with RDQS_t for differential RDQS mode option. RDQS_c[0] is associated with DQ[7:0], RDQS_c[1] is associated with DQ[15:8]. The functionality of RDQS_c pin depends on the MR20 OP[1:0] setting.
WCK2DQI_OSC_CLK	Input	RDQS Interval Oscillator for measuring WCK2DQI: WCK2DQI_OSC_CLK is a clock input acting as the RDQS Interval Oscillator for measuring WCK2DQI time interval. This clock input is optional. It is enabled and required only when defining Verilog macro “MMP_RDQSOSC”.
WCK2DQO_OSC_CLK	Input	RDQS Interval Oscillator for measuring WCK2DQO: WCK2DQO_OSC_CLK is a clock input acting as the RDQS Interval Oscillator for measuring WCK2DQO time interval. This clock input is optional. It is enabled and required only when defining Verilog macro “MMP_RDQSOSC”.
rd_oe_dq	Output	The LPDDR5 DQ output enable signal, which is used for MMP DFI PHY model.

8. Model Parameter Descriptions

The following table provides details on the user adjustable parameters for the Palladium LPDDR5 Memory Model. These parameters may be modified when instantiating an LPDDR5 instance, if necessary, by modifying the HDL parameter declarations and default values which are exposed for access and debug visibility.

Table 6: User Adjustable Parameters

User Adjustable Parameter	Default Value	Description
ROW_ADDR_WIDTH	13	Row address width.
FLAG_3G6G12G24G	0	1: if memory size is 3Gb, 6Gb, 12Gb or 24Gb. 0: if memory size is 2Gb, 4Gb, 8Gb, 16Gb or 32Gb.
tRDQS_PRE	4(*tWCK)	tRDQS_PRE is used to set the static preamble length when MR10 OP[5:4], RDQS_PRE is set to 2'b11. Range: Min 2*tWCK and Max 4*tWCK
RDDQ_WHICH_BYTE	0	RDDQ_WHICH_BYTE is used only in Byte Mode model when the Read DQ Calibration command is issued. It indicates the byte mode DQ bus to be the upper or the lower byte. 0: lower byte 1: upper byte
Latency_mode	1'b0 (x16 device) 1'b1 (x8 device)	LPDDR5 Latency mode (MR0 OP[1])
tWCK2DQI	0	The integer part of timing parameter TWCK2DQI in unit of cycles of WCK. Valid range {0-12}
tWCK2DQI_half	0	The fractional part of timing parameter TWCK2DQI in unit of cycles of WCK. Valid range {0,1}
tWCK2DQO	0	The integer part of timing parameter TWCK2DQO in unit of cycles of WCK. Valid range {0-12}
tWCK2DQO_half	0	The fractional part of timing parameter TWCK2DQO in unit of cycles of WCK. Valid range {0,1}

The following table provides information about the exposed localparams which are NOT user adjustable. On rare occasions, users may need adjust some localparams for their configuration. If this case arises, please contact Cadence emulation or MMP support.

Table 7: Visible Non-User-Adjustable Localparam

Localparam	Default Value	Description
col_addr_width	6	column address width
bkbst_addr_width	8	bank and burst address width summation
data_bits	16 (8 for Byte Mode)	data bus width

Localparam	Default Value	Description
addr_bits	7	CA bus width
byte_width	8	number of bits inside a byte
num_bytes	data_bits/byte_width	number of bytes inside a DQ
mode_reg_width	8	mode register width
total_addr_bits	ROW_ADDR_WIDTH +col_addr_width+bkbst_addr_width	memory capacity address width
banks_burst	(64'b1<<(bkbst_addr_width))	number of banks times number of bursts
rows	((64'b1<<ROW_ADDR_WIDTH) - (64'b1<<(ROW_ADDR_WIDTH-2))*FLAG_3G6G12G24G)	number of rows
cols	(64'b1<<col_addr_width)	number of cols
mem_depth	(banks_burst * rows * cols)	number of DQs
dqfifo_num	5	DQ fifo depth
BL_MWR_16B	16	burst length for masked write in configuration of 16 Banks
BL_MWR_8B	32	burst length for masked write in configuration of 8 Banks
BL_MRR	16	burst length for mode register read
BL_DQFIFO	16	burst length for dq fifo read/write
BL_DQTRAIN	16	burst length for dq calibration training
Latency_mode	0 (1 for Byte Mode)	0: support x16 mode latency; 1: support Byte Mode latency
IO_Width	0 (1 for Byte Mode)	0: support x16 mode IO_width; 1: support Byte Mode IO_width
WRXS	1'b1	0: not support Write X feature; 1: support Write X feature
WDCFS	1'b1	0: not support Data Copy feature; 1: support Data Copy feature
RDCFS	1'b1	0: not support Data Copy feature; 1: support Data Copy feature
Manu_ID	8'b0000_0000	manufacture ID
Ver_ID1	8'b0000_0000	Version ID A
Ver_ID2	8'b0000_0000	Version ID B
MR1_default	8'b0000_0000	default value for mode register MR1
MR2_default	8'b0000_0000	default value for mode register MR2
MR3_default	8'b0000_0110	default value for mode register MR3
MR9_default	8'b0000_0000	default value for mode register MR9
MR10_default	8'b0000_0000	default value for mode register MR10
MR11_default	8'b0000_0000	default value for mode register MR11
MR12_default	8'b0101_0000	default value for mode register MR12
MR13_default	8'b0000_0000	default value for mode register MR13
MR14_default	8'b0101_0000	default value for mode register MR14
MR15_default	8'b0100_0110	default value for mode register MR15
MR16_default	8'b0000_0000	default value for mode register MR16

Localparam	Default Value	Description
MR17_default	8'b0010_1000	default value for mode register MR17
MR18_default	8'b1000_0000	default value for mode register MR18
MR19_default	8'b0000_0000	default value for mode register MR19
MR20_default	8'b0000_0001	default value for mode register MR20
MR21_default	8'b0000_0000	default value for mode register MR21
MR22_default	8'b0000_0000	default value for mode register MR22
MR23_default	8'b0000_0000	default value for mode register MR23
MR25_default	8'b0000_0000	default value for mode register MR25
MR27_default	8'b0000_0000	default value for mode register MR27
MR28_default	8'b0000_0000	default value for mode register MR28
MR29_default	8'b0000_0000	default value for mode register MR29
MR30_default	8'b0000_0000	default value for mode register MR30
MR31_default	8'h55	default value for mode register MR31
MR32_default	8'h55	default value for mode register MR32
MR33_default	8'h5A	default value for mode register MR33
MR34_default	8'h3C	default value for mode register MR34
MR37_default	8'b0000_0000	default value for mode register MR37
MR40_default	8'b0000_0000	default value for mode register MR40
MR41_default	8'b0110_0000	default value for mode register MR41
MR42_default	8'b0000_0000	default value for mode register MR42
MR43_default	8'b0000_0000	default value for mode register MR43
MR44_default	8'b0000_0000	default value for mode register MR44
MR45_default	8'b0000_0000	default value for mode register MR45
MR47_default	8'b0000_0000	default value for mode register MR47
MR48_default	8'b0000_0000	default value for mode register MR48
MR49_default	8'b0000_0000	default value for mode register MR49
MR50_default	8'b0000_0000	default value for mode register MR50
MR51_default	8'b0000_0000	default value for mode register MR51
MR52_default	8'b0000_0000	default value for mode register MR52
MR53_default	8'b0000_0000	default value for mode register MR53
MR54_default	8'b0000_0000	default value for mode register MR54

Note that there are additional exposed localparams in the model HDL that are not described here nor intended to be described here. These additional localparams are exposed for debugging purposes only and will not be described herein.

9. Address mapping

The array of the LPDDR5 model is mapped into the internal memory of the Palladium system. This array is a single two-dimensional array. The mapping of bank, row and column addresses to the internal model array is as follows.

For binary memory densities (2Gb/4Gb/8Gb/16Gb/32Gb)

- Addressing for 4Bank4BankGroup Mode:
  ARRAY_ADDR = {BG[1:0], BA[1:0], ROW[row_bits-1:0], COL[5:0], Burst[3:0]}

- Addressing for 8Bank Mode:
  ARRAY_ADDR = {BA[2:0], ROW[row_bits-1:0], COL[5:0], Burst[4:0]}

- Addressing for 16Bank Mode:
  ARRAY_ADDR = {BA[3:0], ROW[row_bits-1:0], COL[5:0], Burst[3:0]}

For non-binary memory densities (3Gb/6Gb/12Gb/24Gb)

- Addressing for 4Bank4BankGroup Mode:
  *ARRAY_ADDR =(((BG*banks+BA)*rows+ROW)*cols+COL)*burs+Burst[3:0]*

- Addressing for 8Bank Mode:
  *ARRAY_ADDR =((BA*rows+ROW)*cols+COL)*burs+Burst[4:0]*

- Addressing for 16Bank Mode:
  *ARRAY_ADDR =((BA*rows+ROW)*cols+COL)*burs+Burst[3:0]*

('banks' is the total number of banks per bank group; 'rows' is the total number of rows per bank; 'cols' is the total number of columns per row; 'burs' is the total number of burst per column)

This information is required if the memory needs to be preloaded with user data.

The array name in the model hierarchy is: memcore

Due to the constrained depth of the internal memory in the Palladium system, some LPDDR5 models with large memory size are implemented with several smaller memcores instead of one memcore.

When users want to load in or dump out data from memcores, the MSB of the memory address is used to choose from memcore0 (MSB=1'b0) and memcore1(MSB=1'b1).

32 Gb x16 LPDDR5 model consists of two 16Gb memcores.
16 Gb x 8 LPDDR5 model consists of two 8 Gb memcores.

When users want to load in or dump out data from memcores, the MSB and MSB-1 of the memory address are used to choose from memcore0 ([MSB:MSB-1]=2'b00), memcore1 ([MSB:MSB-1]=2'b01), memcore2 ([MSB:MSB-1]=2'b10), AND memcore3 ([MSB:MSB-1]=2'b11).

32 Gb x8 LPDDR5 model consists of four 8Gb memcores.
24 Gb x8 LPDDR5 model consists of three 8Gb memcores.

The user can refer to additional examples of memory load and dump in the runtime section of this user guide.

If {row,bank,col} addressing is needed instead of the default {bank,row,col} addressing, add +define+MMP_RBC to the vlan invocation (IXCOM flow) or to the appropriate HDL-ICE synthesis (Classical flow) command. No value is required for MMP_RBC – only the compile phase define. This option is applicable when using the <model>.vp file. In MMP_RBC mode, the mapping of bank, row and column addresses to the internal model array is as follows:

For binary memory densities (2Gb/4Gb/8Gb/16Gb/32Gb)

- Addressing for 4Bank4BankGroup Mode:
  ARRAY_ADDR = { ROW[row_bits-1:0], BG[1:0], BA[1:0], COL[5:0], Burst[3:0]}

- Addressing for 8Bank Mode:
  ARRAY_ADDR = { ROW[row_bits-1:0], BA[2:0], COL[5:0], Burst[4:0]}

- Addressing for 16Bank Mode:
  ARRAY_ADDR = { ROW[row_bits-1:0], BA[3:0], COL[5:0], Burst[3:0]}

For non-binary memory densities (3Gb/6Gb/12Gb/24Gb)

- Addressing for 4Bank4BankGroup Mode:
  *ARRAY_ADDR =(((ROW*bgs+BG)*banks+BA)*cols+COL)*burs+Burst[3:0]*

- Addressing for 8Bank Mode:
  *ARRAY_ADDR =((ROW*banks+BA)*cols+COL)*burs+Burst[4:0]*

- Addressing for 16Bank Mode:
  *ARRAY_ADDR =((ROW*banks+BA)*cols+COL)*burs+Burst[3:0]*

('bgs' is the total number of bank groups; 'banks' is the total number of banks; 'rows' is the total number of rows; 'cols' is the total number of columns; 'burs' is the total number of burst)

10. Register Definitions

In the LPDDR5 specification there are up to 64 registers defined. The LPDDR5 Palladium model implements the following registers.*

Table 8: Registers Implemented in JEDEC models

Name	Address MA<7:0>	Access	Fields and Description							
			OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
MR0	0x00	R	RFU	RFU	Unified NT-ODT Behavior	DMI Ouput Behavior Mode	Optimized Refresh Mode	Enhanced WCK Always On Mode	Latency Mode	RFU
MR1	0x01	R/W	WL				CK Mode	RFU		
			WL				CK Mode			
			WL				CK Mode			
MR2	0x02	W	nWR				RL and nRBTP			
			nWR				RL and nRBTP			
			nWR				RL and nRBTP			
MR3	0x03	R/W	DBI-WR	DBI-RD	WLS	BK/BG ORG		PDDS		
			DBI-WR	DBI-RD	WLS	BK/BG ORG		PDDS		
			DBI-WR	DBI-RD	WLS	BK/BG ORG		PDDS		
MR4	0x04	R	TUF	ZQ Master	ZQUF	Refresh Rate				
MR5	0x05	R	LPDDR5 Manufacturer ID							
MR6	0x06	R	Revision ID-1							
MR7	0x07	R	Revision ID-2							
MR8	0x08	R	IO Width		Density				Type	
MR9	0x09	W	Vendor Specific Test Register							
MR10	0x0A	W	RDQS PST		RDQS PRE		WCK PST	RFU		RPST Mode
			RDQS PST		RDQS PRE		WCK PST			RPST Mode

* JEDEC, “LPDDR5 Spec”, rev JESD209-5, February 2019

Name	Address MA<7:0>	Access	Fields and Description							
			OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
			RDQS PST		RDQS PRE		WCK PST			RPST Mode
MR11	0x0B	R/W	RFU	CA ODT			NT-ODT EN	DQ ODT		
			RFU	CA ODT			NT-ODT EN	DQ ODT		
			RFU	CA ODT			NT-ODT EN	DQ ODT		
MR12	0x0C	R/W	VBS	Vref(CA)						
				Vref(CA)						
				Vref(CA)						
MR13	0x0D	W	Dual VDD2	CBT Mode	DMD	DMI Output Behavior Mode	RFU	VRO	Thermal Offset	
MR14	0x0E	R/W	VDLC	Vref(DQ[7:0])						
				Vref(DQ[7:0])						
				Vref(DQ[7:0])						
MR15	0x0F	R/W	RFU	Vref(DQ[15:8])						
				Vref(DQ[15:8])						
				Vref(DQ[15:8])						
MR16	0x10	R/W	CBT-PH	VRCG	CBT		FSP-OP		FSP-WR	
MR17	0x11	R/W	x8ODTD Upper	X8odtd Lower	ODTD-CA	RFU	ODTD-CK	SOC ODT		
					ODTD-CA		ODTD-CK	SOC ODT		
					ODTD-CA		ODTD-CK	SOC ODT		
MR18	0x12	W	CKR	WCK2CK Leveling	WCK SYNC	WCK ON	RFU	WCK Termination		
					WCK SYNC	WCK ON		WCK Termination		
					WCK SYNC	WCK ON		WCK Termination		
MR19	0x13	R/W	RFU				DVFSQ		DVFSC	
							DVFSQ		DVFSC	
							DVFSQ		DVFSC	
MR20	0x14	W	RDC DQ Mode	RDC DMI Mode	RFU		WCK Mode		RDQS	
							WCK Mode		RDQS	
							WCK Mode		RDQS	
MR21	0x15	R/W	WXS	WXFE	RDCFE	WDCFE	RFU	WXFS	RDCFS	WDCFS
MR22	0x16	W	RECC		WECC		RFU			
MR23	0x17	W	PASR Segment Mask							
MR24	0x18	R/W	DFE SUPPORT	DFE Quantity for Upper Byte			RFU	DFE Quantity for Lower Byte		
				DFE Quantity for Upper Byte				DFE Quantity for Lower Byte		
				DFE Quantity for Upper Byte				DFE Quantity for Lower Byte		

Name	Address MA<7:0>	Access	Fields and Description							
			OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
MR25	0x19	W	Optimized Refresh Enable	PARC	CA BUS TERM	CA BUS TERM	RFU			
MR26	0x1A	R/W	RDQS TFE	RDQS TFS	DCMU1	DCMU0	DCML1	DCML0	DCM Flip	DCM Start/ Stop
MR27	0x1B	R	TRR mode	TRR mode Ban			Unlimited	MAC Value		
MR28	0x1C	W	RFU		ZQ Mode	RFU		ZQ interval	ZQ Stop	ZQ Reset
MR29	0x1D	R	PPR Resource							
MR30	0x1E	W	DCA for Upper byte				DCA for Lower byte			
			DCA for Upper byte				DCA for Lower byte			
			DCA for Upper byte				DCA for Lower byte			
MR31	0x1F	W	Lower-Byte Invert Register for DQ Calibration							
MR32	0x20	W	Upper-Byte Invert Register for DQ Calibration							
MR33	0x21	W	DQ Calibration Pattern “A”							
MR34	0x22	W	DQ Calibration Pattern “B”							
MR35	0x23	R	WCK2DQI Oscillator Count - LSB							
MR36	0x24	R	WCK2DQI Oscillator Count - MSB							
MR37	0x25	R/W	WCK2DQI interval timer run time setting							
MR38	0x26	R	WCK2DQO Oscillator Count - LSB							
MR39	0x27	R	WCK2DQO Oscillator Count - MSB							
MR40	0x28	W	WCK2DQO interval timer run time setting							
MR41	0x29	W	NT DQ ODT			PRRE	PPR Resource			
			NT DQ ODT				PPR Resource			
			NT DQ ODT				PPR Resource			
MR42	0x2A	W	PPR KEY Protection							
MR43	0x2B	R	DBE_flag	SBE_count						
MR44	0x2C	R	Data ECC Syndromes							
MR45	0x2D	R	Data ECC Syndrome S[8]	Error Byte Lane	DMI ECC Syndromes					
MR46	0x2E	W	RFU					FIFO RDQS Training	RDQS Toggle	Enhanced RDQS
MR47:54	0x2F~36	N/A	Serial ID 1~8							

	Applied when FSP=0
	Applied when FSP=1
	Applied when FSP=2

10.1.MR0 (Read only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
RFU	RFU	Unified NT-ODT Behavior	DMI Ouput Behavior	Optimized Refresh Mode	Enhanced WCK Always	Latency Mode	RFU

			Mode		On Mode		

Latency Mode is set to 1'b0 in LPDDR5 x16 model.
Latency Mode is set to 1'b1 in LPDDR5 x8 model.

Enhanced WCK Always-on Mode is an optional feature and it is set to be 1'b0.
Optimized Refresh Mode is an optional feature and it is set to be 1'b0.
DMI Output Behavior Mode is an optional feature and it is set to be 1'b0.
Unified NT-ODT Behavior is an optional feature and it is set to be 1'b0.

10.2.MR1 (Read/Write)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
WL				CK Mode	RFU		

CK field default is set to differential.

WL field default is set to 4 since MR18[7] CKR default set is 1'b1 which is 2:1 WCK:CK ratio.

10.3.MR2 (Write only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
nWR				RL and nRBTP			

RL field default is set to 6 and nRBTP field default is set to 0 since MR18[7] CKR default set is 1'b1 which is 2:1 WCK:CK ratio.

nWR field default is set to 5 since MR18[7] CKR default set is 1'b1 which is 2:1 WCK:CK ratio.

10.4.MR3 (Read/Write)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
DBI-WR	DBI-RD	WLS	BK/BG ORG		PDDS		

PDDS can be written but has no effect in the LPDDR5 model.

BK ORG field default is set to 4Bank/4Bank Group OP[4:3]=2'b00.

WLS field default is set to 1'b0 which is "set A".

DBI-RD and DBI-WR are set by default to be disabled.

DBI-RD is not supported and MR3 OP[6] is ignored in Data Copy command.

10.5.MR4 (Read only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
TUF	ZQ Master	ZQUF	Refresh Rate				

Refresh Rate MR4 OP[4:0] field default is set to 5'b01001.

ZQUF and TUF are default set to be 0.

10.6.MR5 (Read only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
LPDDR5 Manufacturer ID							

Manufacturer ID field default is set to zero.

Visible Non-User-Adjustable localparam Manu_ID sets the default value.

10.7.MR6 (Read only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
Revision ID-1							

Revision ID-1 field is set by default to zero.

Visible Non-User-Adjustable localparam Ver_ID1 sets the default value.

10.8.MR7 (Read only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
Revision ID-2							

Revision ID-2 field is set by default to zero.

Visible Non-User-Adjustable localparam Ver_ID2 sets the default value.

10.9.MR8 (Read only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
IO Width		Density				Type	

Type field is set by default to S16 4B/4G OP[1:0]=2'b01.

Density field is set by default to 2Gb OP[5:2]= 4'b0000.

IO width field is set by default to x16 OP[7:6]=2'b00.

IO width field is set by default to x8 OP[7:6]=2'b01.

10.10. MR9 (Write only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
Vendor Specific Test Register							

Vendor Specific Test Register field is set by default to 8'h0.

10.11. MR10 (Write only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
RDQS PST		RDQS PRE		WCK PST		RFU	RPST Mode

RDQS PST, RDQS PRE fields are both set to a default of 2'b00.

When RDQS PRE MR10 OP[5:4] = 2'b01, User Adjustable Parameters tRDQS_PRE sets tRDQS_PRE value within the range of 2*tWCK ≤ tRDQS_PRE ≤ 4*tWCK.

10.12. MR11 (Read/Write)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
RFU	CA ODT			NT-ODT EN	DQ ODT		

CA ODT, DQ ODT and NT-ODT EN fields are set by default to disabled.

These fields can be written, but they will have no effect on the model.

10.13. MR12 (Read/Write)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
VBS	VREF(CA)						

Vref(CA) is read/write. It can be changed by MRW and through Command Bus Training.

Vref(CA) default value is 7'b1010000.

VBS is write only. Its default value is 0.

When MRR, DQ[7] will read 0.

10.14. MR13 (Write only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
Dual VDD2	CBT Mode	DMD	DMI Output Behavior mode	RFU	VRO	Thermal Offset	

Thermal Offset field default is set to 2'b00. Thermal Offset can be written, but it will have no effect on the LPDDR5 model.

CBT Mode field default is set to 1'b0.

VRO field default is set to 0. This field can be written, but writing will have no effect on the LPDDR5 model.

DMI Output Behavior mode is set to 1'b0.

DMD field default is set to 0, Data Mask operation enabled. When disabled (OP[5]=1'b1), masked write command is illegal.

Dual VDD2 field default is set to 0. When Dual VDD2 field is set to 1'b1, MR19 OP[1:0] setting is ignored and low power mode cannot be entered.

10.15. MR14 (Read/Write)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
VDLC	V_{REF}(DQ[7:0])						

Vref DQ[7:0] field default is set to 50% of VDDQ OP[6:0]=7'b1010000. This field can be written and read. But it will have no effect on the LPDDR5 model.

VDLC field default is set to 1'b0.

VDLC can be written and read, but writing will have no effect on the LPDDR5 model.

10.16. MR15 (Read/Write)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
RFU	V_{REF}(DQ[15:8])						

Vref DQ[15:8] field default is set to 50% of VDDQ OP[6:0]=7'b1010000. This field can be written and read, but it will have no effect on the LPDDR5 model.

10.17. MR16 (Read/Write)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
CBT-PH	VRCG	CBT		FSP-OP		FSP-WR	

CBT-PH field is set by default to 1'b0.

VRCG is set by default to 0. VRCG can be read and written, but writing will have no effect on the LPDDR5 model.

CBT field is set by default to Normal Operation OP[5:4]=2'b00.

FSP-WR and FSP-OP fields are set to a default with Frequency-Set-Point[0] at 2'b00.

10.18. MR17 (Read/Write)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
x8ODTD Upper	x8ODTD Lower	ODTD-CA	RFU	ODTD-CK	SOC ODT		

SOC ODT, x8ODTD Upper and x8ODTD Lower fields are set to a default of 0.

ODTC-CK and ODTD-CA fields are set by default to be 1'b1.

MR17 can be written, but writing will have no effect on the LPDDR5 model.

10.19. MR18 (Write only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]

CKR	WCK2CK Leveling	RFU	WCK ON	WCK_FM	WCK Termination

WCK Termination field is set by default to 3'b000, ODT disabled. This field can be written, but writing will have no effect on the LPDDR5 model.

WCK_FM(WCK Frequency Mode) is set by default to 0, but there is no High/Low frequency mode in this model.

WCK ON field is set by default to 0 for WCK Always On Mode disabled.

WCK SYNC field default is set to 0. It can be written, but writing will not affect the LPDDR5 model.

WCK2CK Leveling field is set by default to be 0 (disabled).

CKR field is set to by default to be 1 which is 2:1 ration.

10.20. MR19 (Read/Write)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
RFU				DVFSQ		DVFSC	

DVFSQ field is set to a default value of 0. This field can be written, but writing will have no effect on the LPDDR5 model.

DVFSC field is set to a default value of 0. When MR13 OP[7]=1'b1, DVFSC field setting is ignored. Low power mode cannot be entered.

10.21. MR20 (Write only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
RDC DQ Mode	RDC DMI Mode	RFU		WCK mode		RDQS	

RDQS field is set by default to 2'b01. RDQS_t enabled, RDQS_c disabled.

WCK mode is set by default to 2'b00, differential.

When MR20 OP[3:2]=2'b01, MR20 OP[1:0] should set to 2'b01.

When MR20 OP[3:2]=2'b10, MR20 OP[1:0] should set to 2'b11.

10.22. MR21 (Read/Write)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
RFU	WXFE	RDCFE	WDCFE	RFU	WXFS	RDCFS	WDCFS

LPDDR5 model supports the write data copy function with MR21 OP[0]=1'b1, read-only. WDCFE field MR21 OP[4], write-only, is set to a default value of 1'b0, disabled.

LPDDR5 model supports the read data copy function with MR21 OP[1]=1'b1, read-only. RDCFE field MR21 OP[5], write-only, is set to a default value of 1'b0, disabled.

LPDDR5 model supports the write X function with MR21 OP[2]=1'b1, read-only. WXFE field MR21 OP[6], write-only, is set to a default value of 1'b0, disabled.

10.23. MR22 (Write Only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
RECC		WECC		RFU			

LPDDR5 Read Link ECC is enabled by set OP[7:6] to 2'b01, and disabled by set OP[7:6] to 2'b00(default).

LPDDR5 Write Link ECC is enabled by set OP[5:4] to 2'b01, and disabled by set OP[5:4] to 2'b00(default).

10.24. MR23 (Write only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
PASR Segment Mask							

MR23 field is set by default to 8'd0. This field can be written, but writing will have no effect on the LPDDR5 model.

10.25. MR24 (Read/Write)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
DFE SUPPORT	DFE Quantity for Upper Byte			RFU	DFE Quantity for Lower Byte		
	DFE Quantity for Upper Byte				DFE Quantity for Lower Byte		
	DFE Quantity for Upper Byte				DFE Quantity for Lower Byte		

MR24[7:0] is set by default to be 8'b0000_0000.

DFE (Decision Feedback Equalization) is an option feature which is not implemented.

10.26. MR25 (Write only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
RFU	RFU	CA BUS TERM	CK BUS TERM	RFU	RFU	RFU	RFU

CK BUS TERM and CA UT TERM fields are set by default to be 0. These fields can be written, but will have no effect on the LPDDR5 model.

10.27. MR26 (Read/Write)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
RDQSTFE	RDQSTFS	DCMU1	DCMU0	DCML1	DCML0	DCM Flip	DCM Start/ Stop

MR26[7:0] is set by default to be 8'b0000_0000.

MR26 is only set to meet the specification, but there is no real function implemented

10.28. MR27 (Read only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
TRR mode		TRR mode Ban					Unlimited

MR27[7:0] is set by default to be 8'b0000_0000.

TRR (Target Row Refresh) is an optional feature which is not implemented.

10.29. MR28 (Write only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
RFU		**ZQ Mode**	**RFU**	**ZQ Interval**		**ZQ Stop**	**ZQ-Reset**

ZQ stop and ZE-Reset and ZQ Mode fields are set by default to be 0. These fields can be written, but it will have no effect on the LPDDR5 model.

10.30. MR29 (Read only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
PPR Resource Bank 7~0							

MR29[7:0] is set by default to be 8'b0000_0000.

PPR (Post Package Repair) is an optional feature which is not implemented.

10.31. MR30 (Write only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
DCA for Upper byte				**DCA for Lower byte**			

MR30[7:0] is set by default to be 8'b0000_0000.

DCA (Dynamic Cycle Adjustment) is not supported.

10.32. MR31 (Write only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
Lower-Byte Invert Register for DQ Calibration							

MR31 field is set by default to 8'h55. DBI is not applied to RDC command.

10.33. MR32 (Write only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
Upper-Byte Invert Register for DQ Calibration							

MR32 field is set by default to 8'h55. DBI is not applied to RDC command.

10.34. MR33 (Write only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
DQ Calibration Pattern "A"							

MR33 field is set by default to 8'h5A. DBI is not applied to RDC command.

10.35. MR34 (Write only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
DQ Calibration Pattern "B"							

MR34 field is set by default to 8'h3C. DBI is not applied to RDC command.

10.36. MR35 (Read only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
WCK2DQI Oscillator Count - LSB							

If the macro "MMP_RDQSOSC" is not defined, MR35 is set to zero.

When the macro "MMP_RDQSOSC" is defined, the RDQS Internal Oscillator is enabled in the LPDDR5 model. MR35 reports the LSB bits of the DRAM RDQS Oscillator count for measuring WCK2DQI interval.

10.37. MR36 (Read only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
WCK2DQI Oscillator Count - MSB							

If the macro "MMP_RDQSOSC" is not defined, MR36 is set to zero.

When the macro "MMP_RDQSOSC" is defined, the RDQS Internal Oscillator is enabled in the LPDDR5 model. MR36 reports the MSB bits of the DRAM RDQS Oscillator count for measuring WCK2DQI interval.

10.38. MR37 (Read/Write)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
WCK2DQI interval timer run time setting							

When MR37 is OP[7:0]=8'h00, the MPC command [Stop WCK2DQI Interval Oscillator] can stop the RDQS Interval Oscillator.

When MR37 is non-zero the MPC command [Stop WCK2DQI Interval Oscillator] is considered as illegal. When MR37 is non-zero, the WCK2DQI Interval Oscillator automatically stops at the MR37*16th CK clock cycle after the MPC command [Start WCK2DQI Interval Oscillator]. The macro of "MMP_RDQSOSC" should be defined to enable the RDQS Interval Oscillator function.

10.39. MR38 (Read only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]

WCK2DQO Oscillator Count - LSB

If the macro “MMP_RDQSOSC” is not defined, MR38 is set to zero.

When the macro “MMP_RDQSOSC” is defined, the RDQS Internal Oscillator is enabled in the LPDDR5 model. MR38 reports the LSB bits of the DRAM DQS Oscillator count for measuring WCK2DQO interval.

10.40. MR39 (Read only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
WCK2DQO Oscillator Count - MSB							

If the macro “MMP_RDQSOSC” is not defined, MR39 is set to zero.

When the macro “MMP_RDQSOSC” is defined, the RDQS Internal Oscillator is enabled in the LPDDR5 model. MR39 reports the MSB bits of the DRAM DQS Oscillator count for measuring WCK2DQO interval.

10.41. MR40 (Read/Write)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
WCK2DQO interval timer run time setting							

When MR40 is OP[7:0]=8’h00, the MPC command [Stop WCK2DQO Interval Oscillator] can stop the RDQS Interval Oscillator.

When MR40 is non-zero, the MPC command [Stop WCK2DQO Interval Oscillator] is considered as illegal. When MR40 is non-zero, WCK2DQO Interval Oscillator automatically stops at the MR40*16th CK clock cycle after the MPC command [Start WCK2DQO Interval Oscillator]. The macro of “MMP_RDQSOSC” should be defined to enable the RDQS Interval Oscillator function.

.

10.42. MR41 (Write only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
NT DQ ODT			PRRE	PPR Resource			

MR41[7:0] is set by default to be 8’b0000_0000.

PPR (Post Package Repair) is an optional feature which is not implemented.

10.43. MR42 (Write only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
PPR KEY Protection							

MR42[7:0] is set by default to be 8’b0000_0000.

PPR (Post Package Repair) is an optional feature which is not implemented.

10.44. MR43 (Read only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
DBE_flag	SBE_count						

When OP[7] = 1'b0, No double-bit errors have occurred on the DRAM interface.
When OP[7] = 1'b1, One or more double-bit errors have occurred on the DRAM interface.
OP[6:0], The number of times a single-bit error or errors have occurred on the DRAM interface.
NOTE 1 These bits are cleared on power-up, DRAM reset, any Power-Down Exit, and on each read of this mode register. They are also held in the cleared state whenever MR22 OP[5:4]=00b (Link ECC disabled). Because time is required for the DRAM to perform the clear on reads, the delay after reading this mode register should be tMRR + tMRW (rather than simply tMRR as required on most registers).
NOTE 2 The DRAM should disable detection & recording of ECC errors during Write-FIFO commands, as these are used for training purposes and some errors are expected during training.
NOTE 3 In a x16 DRAM device, up to 4 single-bit errors could potentially be detected in a single BL16 burst (byte 0 data SBE, byte 0 DMI SBE, byte 1 data SBE, and byte 1 DMI SBE). Any combination of these 4 simultaneous errors is considered a single SBE occurrence, and would only increment the SBE_count by 1. A BL32 burst can have up to 2 such SBE occurrences, and could increment the SBE_count by up to 2. In a x8 DRAM device, any simultaneous combination of data SBE and DMI SBE is considered a single SBE occurrence, and would only increment the SBE_count by 1 (up to 2 in a BL32 burst).
NOTE 4 In x16 mode, errors from either interface byte are stored in a single register per DRAM (SBE_count would be the number of SBE occurrences on BOTH bytes & DBE_flag would indicate a DBE on EITHER byte). In x8 mode, errors are stored for a single interface byte and MRR returns both copies of this register, each on its corresponding byte lane.
NOTE 5 SBE_count should be a saturating counter.
NOTE 6 Once set, the DBE_flag bit remains set until explicitly cleared by one of the conditions described in note 1 above.

10.45. MR44 (Read only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
Data ECC Syndrome [7:0]							
S[7]	S[6]	S[5]	S[4]	S[3]	S[2]	S[1]	S[0]

Bits 7:0 of the data ECC syndrome from the most recent single-bit error.
NOTE 1 These bits are cleared on power-up, DRAM reset, any Power-Down Exit, and on each read of this mode register. They are also held in the cleared state whenever MR22 OP[5:4]=00B (Link ECC disabled). Because time is required for the DRAM to perform the clear on reads, the delay after reading this mode register should be tMRR + tMRW (rather than simply tMRR as required on most registers).
NOTE 2 The DRAM should disable detection & recording of ECC errors during Write-FIFO commands, as these are used for training purposes and some errors are expected during training.

NOTE 3 In x16 mode, error syndromes from either interface byte are stored in a single register per DRAM. In x8 mode, error syndromes are stored for a single interface byte and MRR returns both copies of this register, each on its corresponding byte lane.
NOTE 4 In x16 mode, it is possible for errors to occur on both bytes simultaneously. In this rare case, only the syndrome from DQ[7:0] should be stored.
NOTE 5 Regardless of detecting a data ECC error or a DMI ECC error, both syndromes should be stored at either error occurrence.

10.46. MR45 (Read only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
Data ECC syndrome[8]	Error Byte Lane	DMI ECC Syndrome					
S[8]	Error Byte Lane	**DS[5]**	**DS[4]**	**DS[3]**	**DS[2]**	**DS[1]**	**DS[0]**

NOTE 1 These bits are cleared on power-up, DRAM reset, any Power-Down Exit, and on each read of this mode register. They are also held in the cleared state whenever MR22 OP[5:4]=00b (Link ECC disabled). Because time is required for the DRAM to perform the clear on reads, the delay after reading this mode register should be tMRR + tMRW (rather than simply tMRR as required on most registers).
NOTE 2 The DRAM should disable detection & recording of ECC errors during Write-FIFO commands, as these are used for training purposes and some errors are expected during training.
NOTE 3 In x16 mode, error syndromes from either interface byte are stored in a single register per DRAM. In x8 mode, error syndromes are stored for a single interface byte and MRR returns both copies of this register, each on its corresponding byte lane.
NOTE 4 In x16 mode, it is possible for errors to occur on both bytes simultaneously. In this rare case, only the error syndromes from byte 0 (DQ[7:0] & DMI0) should be stored, regardless of the error types on the two bytes.
NOTE 5 Regardless of detecting a data ECC error or a DMI ECC error, both syndromes should be stored at either error occurrence.
NOTE 6 In x8 mode, this bit is unused and shall always read back as 0.

10.47. MR46 (Write only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
RFU					FIFO RDQS Training	RDQS Toggle	Enhanced RDQS

MR46[7:0] is set by default to be 8’b0000_0000.

Enhanced RDQS is not supported

10.48. MR47 to MR54 (Read only)

OP[7]	OP[6]	OP[5]	OP[4]	OP[3]	OP[2]	OP[1]	OP[0]
Serial ID-1~8							

MR47 to MR54 are vendor specific

11. Commands

The LPDDR5 model accepts the following commands:

- Deselect
- No operation
- Power Down Entry
- Power Down Exit
- MPC
- Mode Register Write (MRW-1 following MRW-2)
- Mode Register Read
- Activate (ACT-1 following ACT-2)
- Precharge (Per Bank, All Bank)
- Refresh (Per Bank, All Bank)
- CAS
- Read
- Write
- Read32
- Write32
- Mask Write
- Self Refresh Entry
- Self Refresh Exit
- Read FIFO
- Write FIFO
- Read DQ Calibration

The following table shows the command encoding for LPDDR5.

.

Table 9: LPDDR5 Command Encoding

SDRAM COMMAND	DDR COMMAND PINS								CK edge
	CS	CA0	CA1	CA2	CA3	CA4	CA5	CA6	
DES	L	X	X	X	X	X	X	X	R
	X	X	X	X	X	X	X	X	F
NOP	H	L	L	L	L	L	L	L	R
	X	X	X	X	X	X	X	X	F
PDE	H	L	L	L	L	L	L	H	R
	X	X	X	X	X	X	X	X	F
ACT-1	H	H	H	H	R14	R15	R16	R17	R
	X	BA0	BA1	BG0	BG1	R11	R12	R13	F
		BA0	BA1	BA2	BA3				
		BA0	BA1	BA2	V				
ACT-2	H	H	H	L	R7	R8	R9	R10	R
	X	R0	R1	R2	R3	R4	R5	R6	F

SDRAM COMMAND	DDR COMMAND PINS								CK edge
	CS	CA0	CA1	CA2	CA3	CA4	CA5	CA6	
PRE	H	L	L	L	H	H	H	H	R
	X	BA0	BA1	BG0	BG1	V	V	AB	F
		BA0	BA1	BA2	BA3				
		BA0	BA1	BA2	V				
REF	H	L	L	L	H	H	H	L	R
	X	BA0	BA1	BG0	BG1	V	V	AB	F
		BA0	BA1	BA2	BA3				
		BA0	BA1	BA2	V				
WR	H	L	H	H	C0	C3	C4	C5	R
	X	BA0	BA1	BG0	BG1	C1	C2	AP	F
		BA0	BA1	BA2	BA3				
		BA0	BA1	BA2	V				
WR32	H	L	H	H	L	C3	C4	C5	R
	X	BA0	BA1	BG0	BG1	C1	C2	AP	F
		BA0	BA1	BA2	BA3				
MWR	H	L	H	L	C0	C3	C4	C5	R
	X	BA0	BA1	BG0	BG1	C1	C2	AP	F
		BA0	BA1	BA2	BA3				
		BA0	BA1	BA2	V				
RD	H	H	L	L	C0	C3	C4	C5	R
	X	BA0	BA1	BG0	BG1	C1	C2	AP	F
		BA0	BA1	BA2	BA3				
		BA0	BA1	BA2	V				
RD	H	H	L	H	C0	C3	C4	C5	R
	X	BA0	BA1	BG0	BG1	C1	C2	AP	F
		BA0	BA1	BA2	BA3				
CAS	H	L	L	H	H	WS_WR	WS_RD	WS_FAST	R
	X	DC0	DC1	DC2	DC3	WRX	V	B3	F
MPC	H	L	L	L	L	H	H	OP7	R
	X	OP0	OP1	OP2	OP3	OP4	OP5	OP6	F
SRE	H	L	L	L	H	L	H	H	R
	X	V	V	V	V	V	DSM	PD	F
SRX	H	L	L	L	H	L	H	L	R
	X	V	V	V	V	V	V	V	F
MRW-1	H	L	L	L	H	H	L	H	R
	X	MA0	MA1	MA2	MA3	MA4	MA5	MA6	F
MRW-2	H	L	L	L	H	L	L	OP7	R
	X	OP0	OP1	OP2	OP3	OP4	OP5	OP6	F
MRR	H	L	L	L	H	H	L	L	R

SDRAM COMMAND	DDR COMMAND PINS								CK edge
	CS	CA0	CA1	CA2	CA3	CA4	CA5	CA6	
	X	MA0	MA1	MA2	MA3	MA4	MA5	MA6	F
WFF	H	L	L	L	L	L	H	H	R
	X	L	L	L	L	L	L	L	F
RFF	H	L	L	L	L	L	H	L	R
	X	L	L	L	L	L	L	L	F
RDC	H	L	L	L	L	H	L	H	R
	X	L	L	L	L	L	L	L	F

MPC sub-commands:

Table 10: MPC Sub-Commands

Function	Operand	Data
Training Modes	OP[7:0]	10000001B : Start WCK2DQI Interval Oscillator 10000010B : Stop WCK2DQI Interval Oscillator 10000011B : Start WCK2DQO Interval Oscillator 10000100B : Stop WCK2DQO Interval Oscillator 10000101B : ZQ Cal Start 10000110B : ZQ Cal Latch All Others Reserved.

12. DBI and DM Function

The LPDDR5 model supports the Data Mask function for Write operations and the Data Bus Inversion (DBI) function for Write and Read operations. LPDDR5 supports DBI function with a byte granularity. MR3 OP7 and OP6 define DBI-WR and DBI-RD enable bits, and MR13 OP5 defines DMD bit. There are eight possible combinations for the LPDDR5 model with handling of the DM and DBI functions. The Mask Write command only supports BL16 for configuration of 16 Banks/4B4G and BL32 for configuration of 8 Banks.

The below table describes the functional behavior for all combinations.*

Table 11: Functional Behavior for LPDDR5 DM and DBI

DM Function	Write DBI Function	Read DBI Function	DMI Signal during Write Command	DMI Signal during Masked Write Command	DMI Signal During Read Command
Disable	Disable	Disable	Note:1	Note:1,3	Note:2
Disable	Enable	Disable	Note:4	Note:3	Note:2
Disable	Disable	Enable	Note:1	Note:3	Note:5
Disable	Enable	Enable	Note:4	Note:3	Note:5
Enable	Disable	Disable	Note:6	Note:7	Note:2
Enable	Enable	Disable	Note:4	Note:8	Note:2
Enable	Disable	Enable	Note:6	Note:7	Note:5
Enable	Enable	Enable	Note:4	Note:8	Note:5

Notes:
1) DMI input signal is a don't care. DMI input receivers are turned OFF.
2) DMI output drivers are turned OFF.
3) Masked Write Command is not allowed and is considered an illegal command as DM function is disabled.
4) DMI signal is treated as DBI signal and it indicates whether DRAM needs to invert the Write data received on DQs within a byte. The LPDDR5 device inverts Write data received on the DQ inputs in case DMI was sampled HIGH, or leaves the Write data non-inverted in case DMI was sampled LOW.
5) The LPDDR5 DRAM inverts Read data on its DQ outputs associated within a byte and drives DMI signal HIGH when the number of ' 1' data bits within a given byte lane is greater than four; otherwise the DRAM does not invert the read data and drives DMI signal LOW.
6) The LPDDR5 DRAM does not perform any mask operation when it receives Write command. During the Write burst associated with Write command, DMI signal is a don't care and ignored by DRAM.
7) The LPDDR5 DRAM requires an explicit Masked Write command for all masked write operations. DMI signal is treated as DM signal and it indicates which bit time within the burst is to be masked. When DMI signal is HIGH, DRAM masks that bit time across all DQs associated within a byte. All DQ input signals within a byte are don't

* JEDEC, "LPDDR5 Spec", rev JESD209-5, February 2019.

care (either HIGH or LOW) when DMI signal is HIGH. When DMI signal is LOW, the LPDDR5 DRAM does not perform mask operation and data received on DQ input is written to the array.
8) The LPDDR5 DRAM requires an explicit Masked Write command for all masked write operations. The LPDDR5 device masks the Write data received on the DQ inputs if the total count of '1' data bits on DQ[2:7] or DQ[10:15] (for Lower Byte or Upper Byte respectively) is equal to or greater than five, and DMI signal is LOW. Otherwise the LPDDR5 DRAM does not perform mask operation and treats it as a legal DBI pattern; DMI signal is treated as DBI signal and data received on DQ input is written to the array.

13. DQ Training Function

Up to 5 consecutive Write DQ FIFO (WFF) commands with user defined patterns may be issued to the SDRAM to store up to 80 values (BL16*5) per pin that can be read back via the Read DQ FIFO (RFF) command.
Read DQ FIFO is non-destructive to the data captured in the FIFO, so data may be read continuously until it is disturbed by another command to the SDRAM. The Read DQ FIFO pointer is reset internally to FIFO[0] in the DRAM anytime a command other than Read DQ FIFO is received. The Write DQ FIFO pointer is reset internally to FIFO[0] only when SRE command or reset issued.

14. WCK2CK Synchronization

The LPDDR5 model is required to be in WCK2CK synchronization state before the memory controller starts a read DQ burst. The LPDDR5 WCK2CK synchronization process is initiated by a CAS command with the related bit enabled. The CAS command with WCK2CK synchronization should be issued before the write, mask write, read, mode register read, read FIFO, write FIFO and read DQ calibration commands. When there is no ongoing DQ burst, the WCK2CK sync state will be lost. The write X command cannot extend the WCK2CK synchronization. The LPDDR5 model requires a new WCK2CK Sync sequence before starting DQ operation, except for write X command.

During Power Down, the WCK2CK sync state is lost. As the WCK2CK synchronization information is lost with power down entry, the DRAM controller must perform a WCK2CK synchronization sequence after power down exit before DQ operation.

The LPDDR5 model supports WCK free running mode. WCK free running mode is enabled by setting MR18 OP[4] = 1. The WCK2CK synchronization state keeps being turned on until the LPDDR5 model receives a power down, self-refresh power-down or deep-sleep commands or reset. The DRAM controller must keep WCK toggling at its full rate after WCK2CK synchronization regardless of DQ operation.

15. RDQS Interval Oscillator

The LPDDR5 model supports the RDQS Interval Oscillator feature (optional). It can be enabled by defining the Verilog macro “MMP_RDQSOSC” during the compiling phase. It is expected that defining “MMP_RDQSOSC” will slow down the emulation if it is modeled accurately.

The RDQS Interval Oscillator is started by issuing a MPC [Start WCK2DQI/WCK2DQO Interval Oscillator] command with OP[7:0] set as described in the Table 10: MPC Sub-Commands table.

The RDQS Interval Oscillator may be stopped by receiving a MPC [Stop WCK2DQI/WCK2DQO Interval Oscillator] command with OP[7:0] set as described in the Table 10: MPC Sub-Commands. The oscillator may also be stopped after the SDRAM counts to a specific number of CK clocks instructed by the controller (see MR37 and MR40 for more information). If MR37 or MR40 is set to none-zero to automatically stop the RDQS Oscillator, then MPC stop RDQS Interval Oscillator command should not be used (illegal).

When the RDQS Interval Oscillator for measuring WCK2DQI is stopped by either method, the result of the oscillator counter is automatically stored in MR35 and MR36.

When the RDQS Interval Oscillator for measuring WCK2DQO is stopped by either method, the result of the oscillator counter is automatically stored in MR38 and MR39.

The LPDDR5 model requires the user to provide two external dedicated independent clocks as the RDQS Interval Oscillators for measuring the time interval WCK2DQI and WCK2DQO in the LPDDR5 model through the ports of WCK2DQI_OSC_CLK and WCK2DQO_OSC_CLK.

The valid frequency for WCK2DQI_OSC_CLK should be higher than 714MHz (= 1/(2*tWCK2DQImax)=1/(2*700ps)).

The valid frequency for WCK2DQO_OSC_CLK should be higher than 333MHz (= 1/(2*tWCK2DQOmax)=1/(2*1500ps)).

16. Data Copy Low Power Function

LPDDR5 model supports the Data Copy Low Power function. Data Copy Low Power function is an optional feature. It is added on normal Write, Mask Write and/or Read operations with the same latencies. DBI-RD is not supported when Read Data Copy is enabled.

LPDDR5 model supports the write data copy function, users may enable the write data copy function by programming MR21 at runtime (MR21 OP[4]=1'b1, WDCFE, Write Data Copy Function Enable).

LPDDR5 model supports the read data copy function, users may enable the read data copy function by MR21 programming at runtime (MR21 OP[5]=1'b1, RDCFE, Read Data Copy Function Enable).

LPDDR5 Write data copy function is applied to each 8 Byte data granularity per DQ byte. Whenever any data pattern is repeated over 8Byte data, only the reference data is transferred through one DQ link (DQ0 for a lower DQ byte, DQ8 for an upper DQ byte) per DQ byte from a host to a LPDDR5 device. The LPDDR5 device recovers the original 8Byte data by copying the reference data to other 7 DQ data during LPDDR5 device's internal Write operation.

The CAS command delivers Write data copy hit or miss information to LPDDR5 devices with 4bit operands DC0-DC3 (DC0 and DC1 only for BL=16).

LPDDR5 read data copy function is applied to each 8 Byte data granularity per DQ byte. LPDDR5 model includes an internal data comparator logic to determine any data pattern repeatability per 8Byte read data during Read operations. If any data pattern repeatability over 8Byte read data is found, LPDDR5 model returns a reference data DQ link (DQ0 for a lower DQ byte, DQ8 for an upper DQ byte) per DQ byte and a Read data copy hit or miss flag bit through the DMI pin (DM0 for a lower DQ byte, DM1 for an upper DQ byte).

17. Write X

The MMP LPDDR5 model supports the Write X function. The Write X function is an optional feature in the LPDDR5 model.

Users may enable the Write X function by programming MR21 at runtime (MR21 OP[6]=1'b1, WXFE, Write X Function Enable).

A Write X command consists of a CAS command with WRX=1. The Write X enable bit is a sticky bit. After a CAS command with Write X enabled is issued, the subsequent write commands perform Write X operations until a CAS command with Write X disable is issued. When a Write X command is issued, the model will write zeros to the specified address.

Write X function does not utilize WCK or DQ during its operation. Therefore, Write X can be performed to an activated bank without WCK clock or WCK2CK sync operation. Unlike the normal write command, the Write X command does not extend WCK2CK sync state.

18. Enhanced Timing Parameter Accuracy

18.1. Enhanced Timing Parameter Accuracy for TWCK2DQI/TWCK2DQO

The LPDDR5 model supports in a very limited manner some enhanced timing parameter accuracy. The enhanced timing parameter accuracy affects TWCK2DQI and TWCK2DQO. The unit is one half-cycle of WCK. The user is responsible for setting the model's parameters tWCK2DQI and tWCK2DQI_half, tWCK2DQO and tWCK2DQO_half for the two timing parameters.

TWCK2DQI = tWCK2DQI*1 + tWCK2DQI_half*0.5 (unit of tWCK)
TWCK2DQO = tWCK2DQO*1 + tWCK2DQO_half*0.5 (unit of tWCK)

18.2. Runtime Control of Timing Parameters

The following mechanism allows users to change the timing parameters TWCK2DQI and TWCK2DQO at runtime.

Parameter Name	Register Name for Runtime Control
tWCK2DQI	tWCK2DQI_reg
tWCK2DQI_half	tWCK2DQI_half_reg
tWCK2DQO	tWCK2DQO_reg
tWCK2DQO_half	tWCK2DQO_half_reg

The 4 registers are already added the keepNet by default in the RTL module. The user can use 'force' command to change the values of timing parameter after download and during runtime.

force <hierarchy_of_LPDDR5_model>.tWCK2DQI_reg 1
force <hierarchy_of_LPDDR5_model>.tWCK2DQI_half_reg 0
force <hierarchy_of_LPDDR5_model>.tWCK2DQO_reg 1
force <hierarchy_of_LPDDR5_model>.tWCK2DQO_half_reg 0

19. Initialization Sequence

The LPDDR5 model requires that the memory controller follows the initialization sequence as documented in the specification. The sequence includes the following steps:

1. RESET_n asserted, RESET_n de-asserted
2. Init_State0: Power Down Exit command
3. Init_State1: (Optional) MRR or MRW command to set some parameters
4. Init_State2: (Optional) MPC ZQCal start command
5. Init_State3: (Optional) MPC ZQCal latch command
6. Init_State4: (Optional) MRW MR16 CBT mode enter and exit
7. Init_State5: (Optional) Write leveling enter and exit
8. Init_State6: (Optional) Write FIFO(WFF), Read FIFO(RFF) and Read DQ Calibration(RDC)
9. Init_State7: (Optional) MRW to set some parameters.

These steps should be performed in the correct sequence to complete initialization and the init_done signal will be asserted. All the optional steps can be skipped.

20. Model Clocking

The LPDDR5 model has two external clock inputs CK_t/CK_c and WCK_t/WCK_c. The two input clock signals should be in phase with each other.

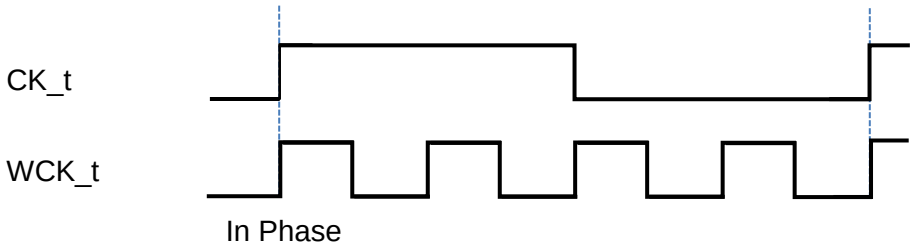


Figure 3: CK and WCK should be in phase when CKR=4:1 (default)

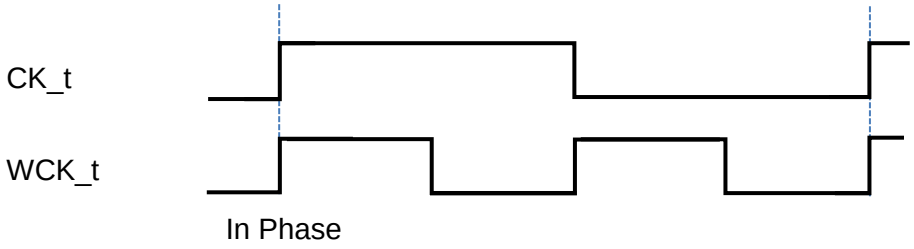


Figure 4: CK and WCK should be in phase when CKR=2:1

21. Compile and Emulation

21.1. Palladium

The model is provided as a protected RTL file(s) (*.vp). The file(s) need to be synthesized prior to the back-end Palladium compilation. An example of the command for compilation (includin) and run of this model in the IXCOM flow is shown below.

```
ixcom -64 +sv -ua +dut+jedec_lpddr5_4Gb \
      ./jedec_lpddr5_4Gb.vp \
   -incdir <mmp_release_install_path>/utils/cdn_mmp_utils/sv \
      <mmp_release_install_path>/utils/cdn_mmp_utils/sv/cdn_mmp_utils.sv \
      ……

xeDebug -64 --ncsim \
   -sv_lib <mmp_release_install_path>/utils/cdn_mmp_utils/lib/64bit/libMMP_utils.so -- \
      -input auto_xedebug.tcl
```

The script below shows two examples for Palladium classic ICE synthesis:

```
1)
hdlInputFile jedec_lpddr5_4Gb.vp
hdlImport -full -2001 -l qtref
hdlOutputFile -add -f verilog jedec_lpddr5_4Gb.vg
hdlSynthesize -memory -keepVhdlCase -keepRtlSymbol -keepAllFlipFlop  jedec_lpddr5_4Gb
……

2)
vavlog jedec_lpddr5_4Gb.vp

vaelab -keepRtlSymbol -keepAllFlipFlop -outputVlog jedec_lpddr5_4Gb.vg jedec_lpddr5_4Gb
……
```

NOTE: It is common for Palladium flows to require –keepallFlipFlop since it removes optimizations which are in place by default. For example, without –keepAllFlipFlop, HDL-ICE can remove flops with constant inputs and merge equivalent FF. The picture above is modified a bit when ICE ATB mode (–atb) is used since then a constant input FF is only optimized out when there is no initial value for it or the initial value is the same as the constant input value.

It is also common for Palladium flows to require –keepRtlSymbol. This option enables the HDL Compiler to keep original VHDL RTL symbols, such as ".", whenever possible. In other words, it maps VHDL RTL signal name a.b to the netlist entry, \a.b. Without this modifier, the signal name would otherwise be converted to a_b in the netlist.

If the recommended compile script includes the aforementioned options, the user must include them to avoid affecting functionality of the design.

21.2. Protium

Starting with Protium 19.11 release, IXCOM is supported for Protium X1. IXCOM is a transaction-based acceleration (TBA) tool for Protium X1. This tool enables evaluation of non-synthesizable part of the design on Xcelium simulator and the RTL part of the design on Protium prototyping platform. IXCOM involves xeCompile to compile the HDL for FPGA prototyping and xrun for non-synthesizable testbench.

21.2.1. In-Circuit Emulation (ICE) Flow

Protium ICE compilation is identical to Palladium ICE compilation, except ATB mode. ATB constructs are not supported in Protium. Refer to *Protium User Guide* for more information.

There are two options supported in Protium ICE compilation. See the examples below for details. The first example shows the hdl* command defined in the compile script file (for example it's called *compile.qel* below) and use xeCompile to do the RTL compile and synthesis. The second example shows using vavlog/vaelab commands to do the same thing as first example.

1) compile.qel:

```
hdlLang verilog
hdlInputFile –add jedec_lpddr5_*Gb.vp
hdlInputFile -add <selected_wrapper>.v
hdlImport          -full -2001
hdlOutputFile  -add -f verilog <selected_wrapper>.vg
hdlSynthesize  -memory -keepRtlSymbol -keepAllFlipFlop \
                    <selected_wrapper>
```

2)

```
vavlog   jedec_lpddr5_*Gb.vp \
                    <selected_wrapper>.v

vaelab   -keepRtlSymbol -keepAllFlipFlop \
                    -outputVlog <selected_wrapper>.vg
```

21.2.2. IXCOM Stub and Check (SC) Flow

The IXCOM Stub and Check flow is available on both Protium X1 and Protium S1 platforms. IXCOM Stub and Check flow enables use of constructs which are synthesizable in IXCOM flow but not in ICE flow. The IXCOM SC flow can help to bring up a design on Protium hardware.

Here is the step-by-step guideline to bring up design on IXCOM SC flow:

1) Stub out all non-RTL code in the design; use *+moduleStub* to stub out verification modules
2) Use *+no_sva* and *+no_psl* to disable assertions.

3) The compile option script file should be present in the compilation flow. The first line of the file should be:

   "precompileOption -add stubIXCOM"

4) **MMP_IXCOM_SC** Verilog macro is required for Protium IXCOM SC flow. User must define this macro for MMP models during compile stage. Here are the examples:

   hdlInputFile +define+**MMP_IXCOM_SC**

   or

   vlan +define+**MMP_IXCOM_SC**

For more detailed information, please refer to "IXCOM Stub and Check Flow" section in the *Protium User Guide*.

21.2.3. IXCOM Flow

A Protium IXCOM flow is only supported on Protium X1 platforms. During the Protium IXCOM X1 flow, the files need to be synthesized prior to the back-end Protium IXCOM compile. An example of the command for compilation (including synthesis) and run of this model in the IXCOM flow is shown below.

```
ixcom –ptm +sv \
        +dut+<selected_wrapper> \
        jedec_lpddr5_*Gb.vp \
        <selected_wrapper>.v \
        <your testbench>.v \
        -top <your testbench top> \
        -incdir ../../../utils/cdn_mmp_utils/sv \
    ../../../utils/cdn_mmp_utils/sv/cdn_mmp_utils.sv \
        ……

xrun -R -64 \
   -sv_lib ../../../utils/cdn_mmp_utils/lib/64bit/libMMP_utils.so \
   -input ptmRun.tcl \
   +ptm_lock_script=./ptm_lock.qel
```

The runtime script *ptmRun.tcl* collects the run-time commands. *xrun* executes the commands one-by-one as soon as the hardware resource is successfully locked. The Protium hardware resource (blade serial number) is defined in the script file, which must be matched to the *.et3config file used during the compile stage.

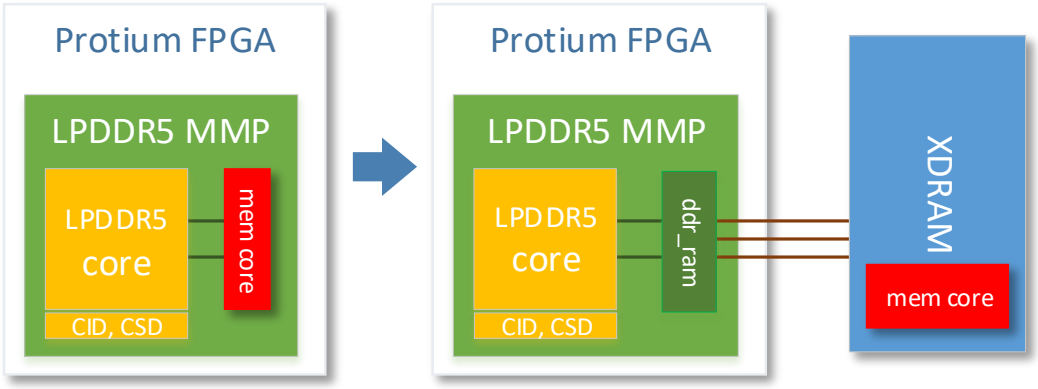
Here is an example for the lock script (*ptm_lock.qel*)

```
system -set 184190096
```

where **184190096** is the serial number of the target blade allocated.

21.2.4. XDRAM Support

The LPDDR5 model supports an XDRAM interface on the Protium platform. The memory core storage inside LPDDR5 model is moved to the XDRAM daughter card once the XDRAM interface is enabled. The block diagram is shown below. LPDDR5supports up to 16GB configuration wrappers with XDRAM interface. User can use MMP_LPDDR5_XDRAM macro to enable the feature.



Enable the XDRAM interface

The user can, during the RTL compilation stage, specify the Verilog macro using +define+MMP_LPDDR5_XDRAM in order to enable the XDRAM interface in the LPDDR5 model.

ICE flow:

```
hdlLang verilog
hdlInputFile –add jedec_lpddr5_*Gb.vp
hdlInputFile -add <selected_wrapper>.v
hdlInputFile -add <ptm_install_path>/share/vxe/ExtCardBitFiles/MEMORY_CARD_D16G_V2
/MMP_PTM/ptm_mmp_pkg/mmp/sram/rtl/ddr_ram.v
hdlInputFile +define+MMP_LPDDR5_XDRAM
hdlImport          -full -2001

or

vavlog   jedec_lpddr5_*Gb.vp \
                   <selected_wrapper>.v \
<ptm_install_path>/share/vxe/ExtCardBitFiles/MEMORY_CARD_D16G_V2
/MMP_PTM/ptm_mmp_pkg/mmp/sram/rtl/ddr_ram.v \
                   +define+MMP_LPDDR5_XDRAM
```

IXCOM SC and IXCOM flow:

```
vlan <ptm_install_path>/share/vxe/ExtCardBitFiles/MEMORY_CARD_D16G_V2
/MMP_PTM/ptm_mmp_pkg/mmp/sram/rtl/ddr_ram.v
vlan +define+MMP_LPDDR5_XDRAM
```

For the details of Protium black-box flow and the XDRAM architectures please review the *Protium User Guide* and *XDRAM3 (39RA342K) User Guide*.

22. Runtime

If the user wishes to preload the memory array, the path to model instance can be found in the Palladium working directory in dbFiles/*mpart.

Example paths for 2/3/4/6/8/12/16/24Gb x16 models and 2/3/4/6/8/12Gb x8 models:

```
stb.dut.memcore
```

Example paths for 32Gb x16 model:

```
stb.dut.memcore0
stb.dut.memcore1
```

Example paths for 32Gb x8 model:

```
stb.dut.memcore0
stb.dut.memcore1
stb.dut.memcore2
stb.dut.memcore3
```

Preloading example:

```
memory –load %readmemh stb.dut.memcore –file mem_load.dat
memory -dump %readmemh stb.dut.memcore -file mem_dump.dat
```

Latency Mode Setting

By default, the ‘Latency Mode’ (MR0 OP[1]) is 1’b0 (X16 mode latency) for LPDDR5 x16 models; the ‘Latency Mode’ (MR0 OP[1]) is 1’b1 (Byte mode latency) for LPDDR5 x8 models.

In the model, there is a user configurable parameter ‘Latency_mode’. The user can change this parameter to configure the ‘Latency Mode’ when doing the LPDDR5 model instantiation.

The LPDDR5 model also provides runtime configurable ability for the ‘Latency Mode’. The register ‘latency_mode_reg’ default value is from the parameter ‘Latency_mode’. The user can force the register ‘latency_mode_reg’ to change ‘Latency Mode’ at runtime. There is default ‘keep_net’ in the model for this register and no recompile needed.

The example for changing ‘Latency Mode’ to ‘Byte mode latency’ is as below:

```
force <hierarchy_of_LPDDR5_model>.latency_mode_reg  1
```

23. SWI Smart Memory Interface

This LPDDR5 model supports a fixed, Verilog macro controlled interface, or “hooks”, to one of the Smart Memory components in Cadence’s Software Integrator (SWI) product. Software Integrator (SWI) is a support library that is part of Cadence’s Hybrid Solution--a multi-tool system level solution that runs in a hybrid PXP + IES simulation mode and targets the increasing number of SoC performance conscious projects requiring the booting of commercial operating systems and the running of complex software-driven tests against an accurate model of the SoC prior to tapeout.

For additional details about the SWI product at large please consult the SWI product documentation which includes a user guide. This documentation can be accessed via support.cadence.com and is located on the Product Pages/Product Manuals link where SWI 13.1 is located with other Functional Verification products.

The user of the SWI solution who is integrating this core model to the corresponding SWI Smart Memory component side should define the Verilog macro “MMP_SM” to enable the Smart Memory interface. This will enable the portion of the interface that resides in the MMP model core, thus completing access to the implemented SWI Smart Memory functionality.

Table 12: SWI Smart Memory Verilog Define

`define Macro purpose	Possible `define Values
Set the Smart Memory interface to compile “in” as part of the memory model	MMP_SM

The SWI Smart Memory interface includes the signals shown in the table below. It is outside the MMP scope to treat the integration of the MMP model into a hybrid solution. For additional details, please consult the SWI documentation and other Hybrid Solution documentation.

Table 13: SWI Smart Memory Interface Signals

NAME	DECLARATION	DESCRIPTION
sm_fastclk	output sm_fastclk	Smart Memory interface clock
sm_raddr	output [(total_addr_bits-1):0] sm_raddr	Smart Memory read address
sm_re	output sm_re	Smart Memory read enable
sm_raddr_en	output sm_raddr_en	Smart Memory read address enable
sm_dout	input [data_bits-1:0] sm_dout	Smart Memory read data
sm_waddr	output [(total_addr_bits-1):0] sm_waddr	Smart Memory write address
sm_we	output sm_we	Smart Memory write enable
sm_waddr_en	output sm_waddr_en	Smart Memory write address enable
sm_din	output [data_bits-1:0] sm_din	Smart Memory write data
sm_bw	output [data_bits-1:0] sm_bw	Smart Memory data mask
verbosity	input [3:0] verbosity	Smart Memory debug info display level control

The Smart Memory interface includes a user adjustable parameter that passes user data from wrapper layers that are external to this MMP model into MMP model. The single 64-bit parameter is subdivided by field to accommodate data as shown in the table below. This parameter defaults to a value of '0' and is managed by the SWI product Smart Memory component. For additional details, please consult the SWI documentation and other Hybrid Solution documentation.

Table 14: SWI Smart Memory User Adjustable Parameters

PARAMETER	**DESCRIPTION**
parameter [63:0] SMConfigSpecific_UserData	Smart Memory User Data Passing
FIELD	**DESCRIPTION**
[2:0]	Used for specifying device/channel/subpart
[63:61]	Extension value of total_addr_bits

The Smart Memory interface includes non-user adjustable localparams and parameters as shown in the table below. Note that a localparam of the same name (total_addr_bits) but of different size is part of the standard MMP core model.

Table 15: SWI Smart Memory Non-User Adjustable Parameters

LOCALPARAM	**VALUE**	**DESCRIPTION**
total_addr_bits	bank_addr_width+row_addr_width +col_addr_width + SMConfigSpecific_UserData[63:61]	Memory capacity width

The SWI Smart Memory interface has dependencies on the inclusion of external file(s). For additional details about purpose and content of any Smart Memory related external files, please consult the SWI documentation and other Hybrid Solution documentation.

```
`ifdef MMP_SM
   `include "cdn_sm_mapDRBCToLinAdr.vh"
`endif
```

NOTE: The LPDDR5 Write X and Data Copy features are NOT supported for the Smart Memory Interface.

24. Link ECC

Link ECC is an optional feature which is defined in the official released JEDEC specification (JESD209-5.pdf). For Link ECC functional details and protocol definition, please refer to JEDEC specification JESD209-5.pdf. For Link ECC enable and disable mode register configuration, please refer to section 10.23. For Link ECC enabled error reporting, please refer to sections 10.39, 10.40, and 10.41.

For pinout description while Link ECC is enabled, please refer to the “I/O Signal Description” table above.
Link ECC feature does not support Error Correction.
For details about Link ECC feature support, please refer to the table below.

ITEM	LPDDR5 FEATURE	SUPPORT	DESCRIPTION
1	Error Correction	No	The JED209-5 specification does not define the particular cycle to do error correction. In the worst case of continuous WRITE OPERATIONs, there will not be enough time to correct the data and save the data into the memory core if there are errors detected.
2	Error Reporting	Yes, with limits	In the case of Data_bit 16 and Burst Length 16, two sets of data Link ECC syndromes and two sets of DMI Link ECC syndromes are generated at the same time. Mode Register 44/45 can only record one syndrome of these four sets of syndromes. (Details please refer to 10.40 and 10.41 MR44/45 NOTES) In the case of Data_bit 16 and Burst Length 32, four sets of data Link ECC syndromes and four sets of DMI Link ECC syndromes are generated at the same time. Mode Register 44/45 can only record a maximum of two syndromes of these eight sets of syndromes. (Details please refer to 10.40 and 10.41 MR44/45 NOTES)
			In the worst case of (Data_bit 16 AND Burst Length 32) OR (Data_bit 16 AND Burst Length 16), either the data syndromes or DMI syndromes show the double bit error in any of these simultaneous combinations of ECC syndrome sets. The other syndrome will show a single bit error. For these cases, the MR43 will also only show a double bit error and will ignore the other single bit error. The single bit error counter will not be incremented.
3	Write FIFO commands Read FIFO commands READ DQ Calibration Commands	No	Errors are expected to occur during these training commands. Per JESD209-5 specification, the model will not increment the SBE_count or set the DBE_flag in MR43 during these training operations. Per JESD209-5 specification, the model will not save the ECC syndromes in MR44 and MR45 during these training operations.

ITEM	LPDDR5 FEATURE	SUPPORT	DESCRIPTION
4	Write Data Copy command	Yes	N/A
5	WRITE OPERATION DATA ECC generation	Yes	N/A
6	WRITE OPERATION DMI ECC generation	Yes	N/A
7	READ OPERATION DATA ECC generation	Yes	N/A
8	ECC check matrix ENCODING	Yes	N/A
9	ECC check matrix DECODING	Yes	N/A
10	ERROR DETECTION	Yes	N/A
11	MODE REGISTER 22 Configuration to Enable/Disable Link ECC	Yes	N/A
12	MODE REGISTER 43 SBE/DBE	Yes	N/A
13	MODE REGISTER 44/45 reading for most recent SBE	Yes	N/A

Table 16: Link ECC Feature Support

NOTE:
During a READ operation, the LPDDR5 model will either perform DBI encoding (if DBI is enabled), or ECC generation on the data (if ECC is enabled). The operation performed depends on which feature is enabled since READ DBI and Link ECC operations are mutually exclusive.
.

25. Handling WCK in Palladium Memory Models

For writes to a DDR memory, industry datasheets show each WCK edge centered within the corresponding valid period (v0, v1, v2, etc.) of DQ, as in the following diagram.

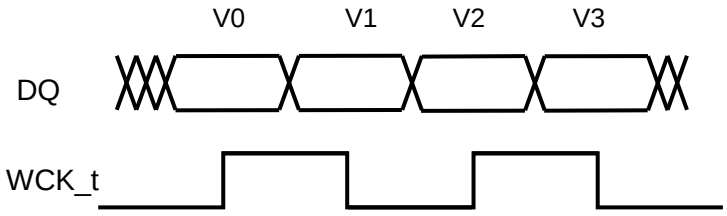


Figure 5: DQ and WCK_t write timing relation in industry

For DDR models provided by Cadence for Palladium, if the design drives DQ and WCK signals with the above timing, the DDR memory will behave correctly. However, to obtain this timing in Palladium, the fastest design clock must toggle twice as frequently as the WCK signal. If this faster clock is not needed for any other reason, the presence of the faster clock will usually cause an unnecessary 2X slowdown in emulation speed. To eliminate the need for a faster clock, you can have the design generate each WCK edge at the end of the corresponding DQ valid period (rather than the middle), as in the following diagram:

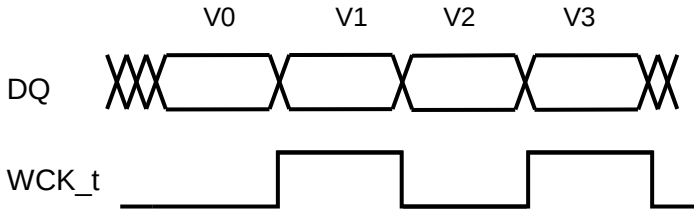


Figure 6: DQ and WCK_t write timing relation in Palladium

Note that the first WCK edge is at the *end* of first valid DQ, not at the beginning.

For reads from the DDR model, the DDR model will drive DQ with the first WCK edge at the *beginning* of the first valid data, not at the end:

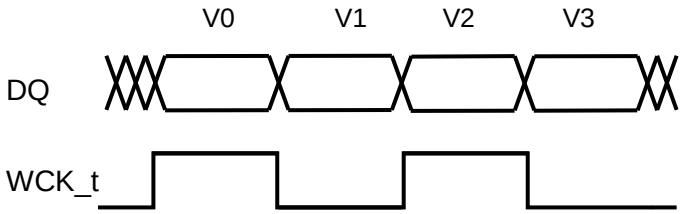


Figure 7: DQ and WCK_t read timing relation in Palladium

The DDR model behaves this way to conform to industry datasheets for DDR memories. The design reading the data from the DDR model must delay the WCK signal, and use the delayed-WCK signal to sample the DQ. A delay of one Q_FDP0B should work fine, even in CAKE 1X mode. If you are using CAKE 1X mode and the DDR clock is the fastest design clock, the DQ signal will change twice per FCLK, and the Q_FDP0B delaying DQS will provide one-half FCLK delay, so that each delayed-DQS edge is at the end of the corresponding data valid period.

To delay the WCK signal, a commonly used approach is to create a special pad cell for WCK that has a Q_FDP0B delay cell inserted on the path that leads from the DDR memory into the design.

The user may insert delays into pad cells (or elsewhere in the design) using the below code example which leverages ixc_pulse, an internal primitive that can be used to access FCLK and to create controlled delay, for IXCOM flow and leverages the Q_FDP0B primitive for delay generation in the Classic ICE flow. For more detailed information about ixc_pulse please reference the *VXE User Guide* section called *Generating Pulses*. There is no need for the user to define IXCOM_UXE for the Verilog macro; it is predefined for the user in IXCOM flow.

```
// Flow independent delay cell
module pxp_fclk_delay (in, out_delay);
input in;
output out_delay;

reg out_delay;

`ifdef IXCOM_UXE
  wire VCC=1'b1;
  ixc_pulse #(1)(Fclk,VCC);
  always @(posedge Fclk)
    out_delay <= in;
`else
  Q_FDP0B fclk_dly (.D(in), .Q(out_delay));
`endif

endmodule
```

26. Debugging

This model has some debugging options, techniques and tips which may assist users with isolating problems.

- For issues which are not model specific, please review the *Memory Model Portfolio FAQ for All Models User Guide.*

- **Debug Signals:**

  The following signals can be monitored to examine command sequence:

  - init_done — asserted when initialization sequence is finished
  - init_STATE — indicates the initialization sequence state
  - wck2ck_sync — indicates the wck2ck synchronization state
  - cmd — indicates the received command
  - mem_dout — memcore output
  - mem_dout_addr — memcore output address
  - wr_dq_in_window_real — memcore input enable for posedge data
  - wr_dq_in_window_real_f — memcore input enable for negedge data
  - mem_din_r — memcore input at WCK_t rising edge
  - mem_din_f: — memcore input at WCK_t falling edge
  - mem_din_addr_real_r — memcore input address at WCK_t rising edge
  - mem_din_addr_real_f — memcore input address at WCK_t falling edge
  - mrr_dq_out_window_real — DQ data read out window
  - rd_dq_out_window_real — DQ data read out window
  - wr_dq_out_window_real — DQ data write in window
  - mwr_dq_out_window_real — DQ data masked write in window
  - rdc_dq_out_window_real — read DQ calibration window
  - rff_dq_out_window_real — read DQ FIFO window
  - wff_dq_in_window_real — write DQ FIFO window
  - cbt_dq_out_window — command bus training DQ output window
  - BK_ORG — indicates LPDDR5 model bank configuration
  - BL — burst length
  - WL — burst write latency
  - RL — burst read latency

- **Golden waveform:** A package with a reference waveform is available. The waveform is the running result of the following command sequence(s):
  (1) Initialization sequence:
  PDX
  MRW WCK_ON=1
  CAS
  MRR
  MPC ZQ Calibration Start
  MPC ZQ Calibration Latch
  MRW CBT Enter
  MRW CBT Exit
  MRW Write Leveling On
  MRW Write Leveling OFF

WFF
RFF
RDC
MRW

(2) Other included commands:

ACT
MWR
WR
RD
WR32
RD32
PRE
REF
SRE
DSM
PDE
PDX
SRX

- **Debug Display:** This MMP memory model has available a built-in debug methodology called MMP Debug Display that is based on the Verilog system task $display. Please see the *Palladium Memory Model Debug Display User Guide* in the release docs directory for additional information.

- **Manual Configuring of this MMP Model Family**

  This MMP model supports manual configuration by accompanying the model mode register or configuration register declarations with synthesis directives, such as keep_net directives, that instruct the compiler to ensure that the relevant nets remain available for runtime forcing. For a general description of this support please see the user guide in the MMP release with path and filename *docs/MMP_FAQ_for_All_Models.pdf*.

  While MMP strongly recommends following protocol-based commands to configure MMP models, MMP recognizes that the design test environment may desire to trade off the risks inherent in streamlining or circumventing the initialization sequence part of the protocol in order to better support some testing environments.

  The following table lists the internal register path and naming along with the specification or datasheet naming for model mode registers or configuration registers that are accompanied by keep_net synthesis directives in support of such manual configuration. ONLY writeable configuration registers or fields are supported thusly. Please read the relevant datasheet for details about individual register behavior and mapping to fields.

Table 17: Writeable Mode Register / Configuration Register Info

Hierarchical RTL Naming for Writeable Configuration Related Registers & Signals	Specification or Vendor Datasheet Naming for Configuration Related Registers	Access
<model_name>.MR1_0_reg	MR1 (Set point 0)	W
<model_name>.MR1_1_reg	MR1 (Set point 1)	W
<model_name>.MR1_2_reg	MR1 (Set point 2)	W
<model_name>.MR2_0_reg	MR2 (Set point 0)	W
<model_name>.MR2_1_reg	MR2 (Set point 1)	W
<model_name>.MR2_2_reg	MR2 (Set point 2)	W
<model_name>.MR3_0_reg	MR3 (Set point 0)	W
<model_name>.MR3_1_reg	MR3 (Set point 1)	W
<model_name>.MR3_2_reg	MR3 (Set point 2)	W
<model_name>.MR9_reg	MR9	W
<model_name>.MR10_0_reg	MR10 (Set point 0)	W
<model_name>.MR10_1_reg	MR10 (Set point 1)	W
<model_name>.MR10_2_reg	MR10 (Set point 2)	W
<model_name>.MR11_0_reg	MR11 (Set point 0)	W
<model_name>.MR11_1_reg	MR11 (Set point 1)	W
<model_name>.MR11_2_reg	MR11 (Set point 2)	W
<model_name>.MR12_0_reg	MR12 (Set point 0)	R/W
<model_name>.MR12_1_reg	MR12 (Set point 1)	R/W
<model_name>.MR12_2_reg	MR12 (Set point 2)	R/W
<model_name>.MR13_reg	MR13	W
<model_name>.MR14_0_reg	MR14 (Set point 0)	R/W
<model_name>.MR14_1_reg	MR14 (Set point 1)	R/W
<model_name>.MR14_2_reg	MR14 (Set point 2)	R/W
<model_name>.MR15_0_reg	MR15 (Set point 0)	R/W
<model_name>.MR15_1_reg	MR15 (Set point 1)	R/W
<model_name>.MR15_2_reg	MR15 (Set point 2)	R/W
<model_name>.MR16_reg	MR16	R/W
<model_name>.MR17_0_reg	MR17 (Set point 0)	W
<model_name>.MR17_1_reg	MR17 (Set point 1)	W
<model_name>.MR17_2_reg	MR17 (Set point 2)	W
<model_name>.MR18_0_reg	MR18 (Set point 0)	W
<model_name>.MR18_1_reg	MR18 (Set point 1)	W
<model_name>.MR18_2_reg	MR18 (Set point 2)	W
<model_name>.MR19_0_reg	MR19 (Set point 0)	W
<model_name>.MR19_1_reg	MR19 (Set point 1)	W
<model_name>.MR19_2_reg	MR19 (Set point 2)	W
<model_name>.MR20_0_reg	MR20 (Set point 0)	W
<model_name>.MR20_1_reg	MR20 (Set point 1)	W
<model_name>.MR20_2_reg	MR20 (Set point 2)	W
<model_name>.MR21_reg	MR21	W
<model_name>.MR23_reg	MR23	W
<model_name>.MR25_reg	MR25	W

<model_name>.MR27_reg	MR27	W
<model_name>.MR28_reg	MR28	W
<model_name>.MR30_reg	MR30	W
<model_name>.MR31_reg	MR32	W
<model_name>.MR32_reg	MR32	W
<model_name>.MR33_reg	MR33	W
<model_name>.MR34_reg	MR34	W
<model_name>.MR37_reg	MR37	W
<model_name>.MR40_reg	MR40	W
<model_name>.MR41_reg	MR41	W
<model_name>.MR42_reg	MR42	W
<model_name>.init_done	[Not Applicable]	1'b1 indicates initialization is complete
<model_name>.latency_mode_reg	Latency mode (MR0 OP[1])	W
<model_name>.tWCK2DQI	integer part of timing parameter TWCK2DQI	W
<model_name>.tWCK2DQI_half	fractional part of timing parameter TWCK2DQI	W
<model_name>.tWCK2DQO	integer part of timing parameter TWCK2DQO	W
<model_name>.tWCK2DQO_half	fractional part of timing parameter TWCK2DQI	W
<model_name>.WL	register for write latency	W
<model_name>.RL	register for read latency	W

Revision History

The following table shows the revision history for this document

Date	Version	Revision
December 2017	0.1	Initial release
January 2018	0.2	Modify header and footer
June 2018	0.3	Add section for Manual configuration
July 2018	0.4	Update for new utility library
July 2018	0.5	Update to draft specification R096v8
September 2018	0.6	Update for SWI SM interface Update the initialization sequence to make the training as optional
May 2019	0.7	Add information for extra output port rd_oe_dq Update the address mapping
June 2019	1.0	Move model to non-beta (MR) state Removed UXE & Palladium XP series references
July 2019	1.1	Update for Latency Mode setting
October 2019	1.2	Adding supported timing TWCK2DQI/TWCK2DQO description
January 2020	1.3	Add Link ECC feature information
April 2020	1.4	Update to JEDEC209-5 Specification