HMIC_S IP

User Guide

(UG042003, V1.14)

30.04.2024

Revisions History

Document Revisions

Version	Date of Release	Revisions	Applicable IP and Corresponding Versions
V1.14	30.04.2024	Initial release.	V1.14

IP Revisions

IP Version	Date of Release	Revisions
V1.14	30.04.2024	Initial release.

About this Manual

Terms and Abbreviations

Terms and Abbreviations	Meaning
APB	Advanced Peripheral Bus
AXI	AdvancedeXtensible Interface
DDR	Double Data Rate
DFI	DDR PHY Interface
LP	Low Power
MC	Memory Controller
MR	Mode Register
MRS	Mode Register Set
PHY	Physical
RD	Read
UI	User Interface
WR	Write
DCD	DDR Command Decode
DCP	DDR3 Command Procedure
HMIC	High performance Memory Interface Controller
IPC	IP Compiler
PDS	Pango Design Suite
PCE	Physical Constraint Editor
UCE	User Constraint Editor

Related Documentation

The following documentation is related to this manual:

1. ***Pango_Design_Suite_Quick_Start_Tutorial***
2. ***Pango_Design_Suite_User_Guide***
3. ***IP_Compiler_User_Guide***
4. ***Simulation_User_Guide***
5. ***User_Constraint_Editor_User_Guide***
6. ***Physical_Constraint_Editor_User_Guide***
7. ***Route_Constraint_Editor_User_Guide***

8. *JESD79-3D, DDR3 SDRAM Standard*
9. *JESD79-2F, DDR2 SDRAM SPECIFICATION*
10. *JESD209B, Low Power Double Data Rate (LPDDR) SDRAM Standard*
11. *DDR PHY Inteface, DFI 3.1 Specification*
12. *UG050012_Titan2 Single Board Hardware Design Guide*
13. *UG040012_Logos2 Single Board Hardware Design User Guide*

Table of Contents

Tables

Figures

Chapter 1 Preface

This chapter describes the scope, structure, and writing standards of this manual to help users quickly find the information they need.

1.1 Introduction of the Manual

This manual serves as the user guide for the DDR3 IP—HMIC_S (High-performance Memory Interface Controller Soft core) IP launched by Pango Microsystems. The content of this manual primarily includes the IP user guide and related information. This manual helps users quickly understand the HMIC_S IP features and usage.

1.2 Writing Standards of the Manual

Table 1-1 Description of Writing Standards

Text	**Rules**
Attention	If users ignore the attention contents, they may suffer adverse consequences or fail to operate successfully due to incorrect actions.
Description	Instructions and tips provided for users.
Recommendation	Recommended settings and instructions for users.

Chapter 2 IP User Guide

This chapter provides a guide on the use of HMIC_S IP, including an introduction to IP, block diagram, IP generation process, Example Design, IP interface description, IP register description, typical applications, instructions and considerations, and IP debugging methods. More details on the design process can be found in the following PDS help documentation.

- "***Pango_Design_Suite_Quick_Start_Tutorial***"
- "***Pango_Design_Suite_User_Guide***"
- "***IP_Compiler_User_Guide***"
- "***Simulation_User_Guide***"

2.1 IP Introduction

HMIC_S IP is a DDR3 IP launched by Pango Microsystems. It is compatible with LPDDR and DDR2. Users can configure and generate the IP modules using the IPC (IP Compiler) in the PDS (Pango Design Suite).

2.1.1 Key Features

The main features of the HMIC_S IP product are as follows.

- Supports DDR3, DDR2 and LPDDR;
- Supports a maximum data width of 72 bits;
- User interface: reduced AXI4 bus interface and APB bus interface;
- Supports configurable low-power modes: Self-Refresh and Power Down;
- Supports up to 1066Mbps data rate for DDR3;
- Supports up to 800Mbps data rate for DDR2;
- Supports up to 400Mbps data rate for LPDDR;
- Burst Length 8 and single Rank;
- Available as a separate PHY use.

2.1.2 Applicable Devices and Packages

Table 2-1 HMIC_S IP Applicable Devices and Packages

Applicable Devices	**Supported Packages**
PG2L25H	MBG325
PG2L50H	MBG324/FBG484
PG2L100H	FBG676/FBG484/MBG324
PG2L100HX	FBG676/FBG484/MBG324
PG2L200H	FBB484/FBB676/FFBG1156
PG2T390H	FFBG900/FFBG676
PG2K400	FFBG900/FFBG676
PG2T70H	FBB484/FBB676
PG2T390HX	FFBG900/FFBG676
PG2T160H	FBB484/FFBG676

2.2 IP Block Diagram

The system block diagram of HMIC_S IP is shown in Figure 2-1. HMIC_S IP supports two operating modes: Controller + PHY, or PHY Only.

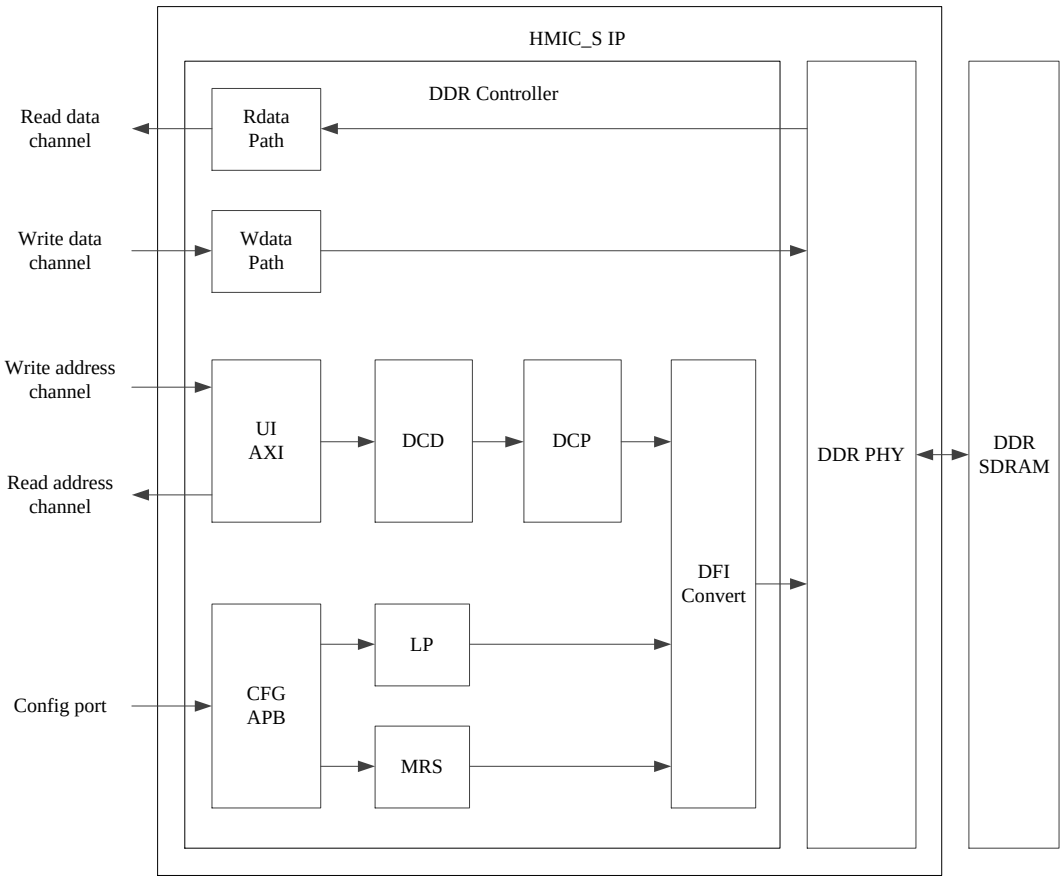

Figure 2-1 HMIC_S IP System Block Diagram

2.2.1 Controller + PHY

In this mode, the IP includes DDR Controller and DDR PHY functions. Users read and write data through the AXI4 interface, and use the APB interface for low power mode and MRS control.

- AXI4 Interface

This interface comprises four parts: write address channel, read address channel, write data channel, and read data channel.

Users initiate read and write operations through the write and read address channels; the commands are parsed into internal controller commands in the UI AXI module; then decomposed into corresponding DDR commands in the DCD (DDR Command Decode) module; timing control based on DDR is implemented in the DCP (DDR3 Command Procedure) module; converted into the DFI interface in the DFI Convert module, transferred to PHY, and finally to the DDR Memory interface.

Write data is passed through the write data channel interface, directly through the Wdata Path module to the DDR PHY, and ultimately transferred to the DDR Memory interface.

Read data from DDR Memory, after being sampled and parsed by the DDR PHY, is synchronized through the Rdata Path module and returned to the user through the read data channel interface.

- Config port

This interface is an APB configuration interface through which users can read the status of DDR SDRAM and implement low power mode and MRS control.

2.2.2 PHY Only

In this mode, users need to design the Controller by their own, and interconnect the PHY with DFI interface to operate the DDR SDRAM.

2.3 IP Generation Process

2.3.1 Module Instantiation

Customized configurations of HMIC_S IP can be completed through the IPC tool, instantiating the

required IP modules. For detailed instructions on using the IPC tool, please refer to "***IP_Compiler_User_Guide***".

The main steps for instantiating the HMIC_S IP module are described below.

2.3.1.1 Selecting IP

Open IPC and click File > Update in the main window to open the Update IP dialog box, where you add the corresponding version of the IP model.

After selecting the FPGAs device type, the Catalog interface displays the loaded IP models. Select the corresponding version of DDR3 Interface under the "System/DDR/Soft" directory. The IP selection path is shown in Figure 2-2. Then set the Pathname and Instance Name on the right side of the page. The project instantiation interface is shown in Figure 2-3.

Attention:

PG2L25H, PG2L50H: The software version must be 2022.1 or above;

PG2L100H, PG2T390H: The software version must be 2021.1-SP7.6 or above;

PG2L200H: The software version must be 2022.2-SP1.2 or above;

PG2L100HX: The software version must be 2023.1 or above;

PG2K400: The software version must be 2023.2 or above;

PG2T70H: The software version must be 2023.2 or above;

PG2T390HX: The software version must be 2022.2-SP6.4 or above;

PG2T160H: For the software version, please consult FAE.

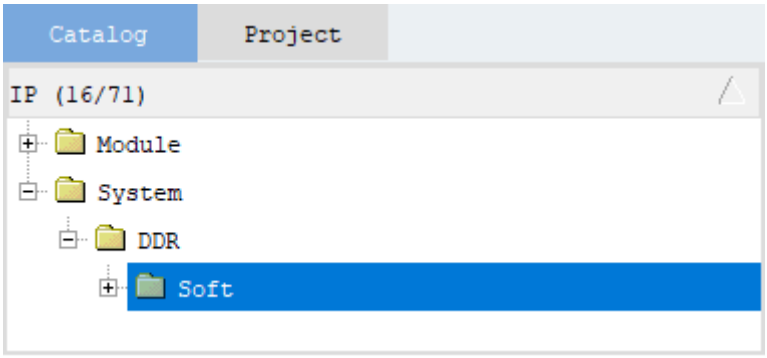

Figure 2-2 HMIC_S IP Path Selection Interface

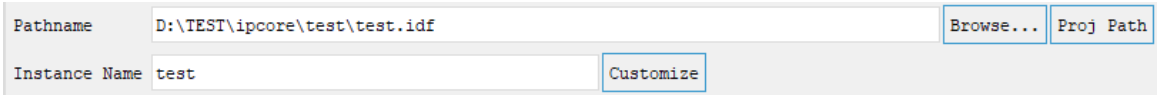

Figure 2-3 Project Instantiation Interface

2.3.1.2 IP Parameter Configuration

After selecting the IP, click <Customize> to enter the HMIC_S IP parameter configuration interface. The left Symbol is the interface block diagram, as shown in Figure 2-4; the Parameter Configuration window is shown on the right side, as shown in Figure 2-5.

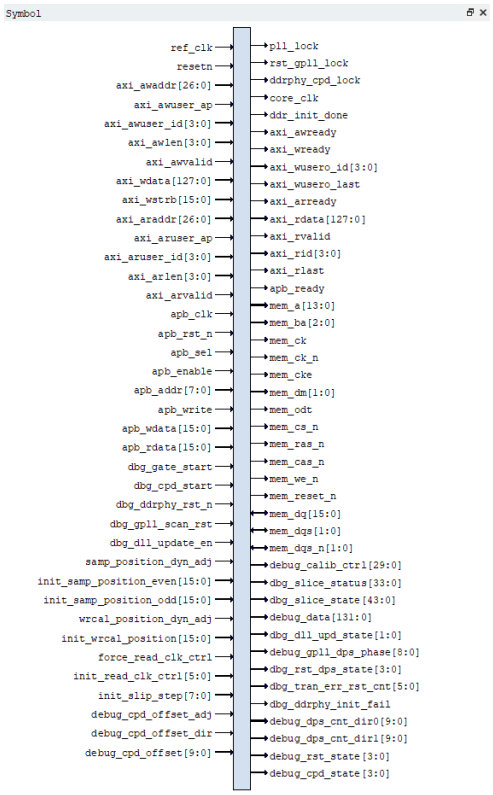

Figure 2-4 HMIC_S IP Interface Block Diagram

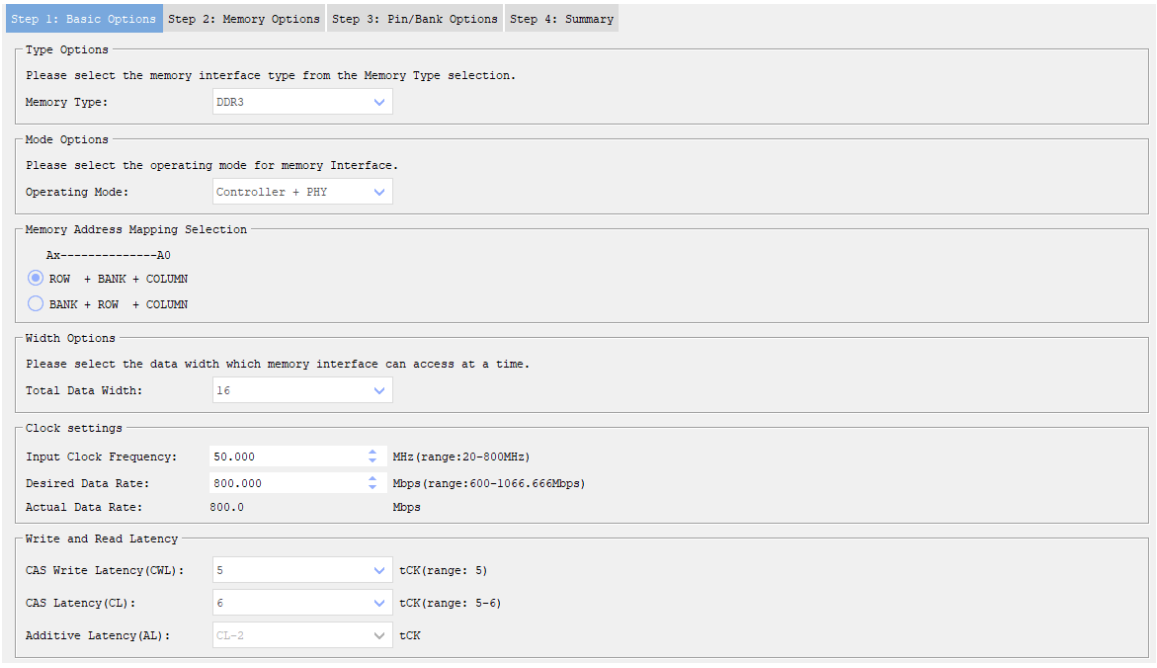

Figure 2-5 HMIC_S IP Parameter Configuration Interface

Attention:

Please be sure to configure the IP parameters in the order of the pages, following Step 1 → Step 2 → Step 3 → Step 4.

Parameter configuration includes four pages, namely Step1: Basic Options, Step 2: Memory Options, Step 3: Pin/Bank Options, Step 4: Summary. The configuration steps for HMIC_S IP are described as follows.

2.3.1.2.1 Step 1: Basic Options

Basic Options is the fundamental configuration page for the IP, as shown in Figure 2-6. Please refer to Table 2-2 for detailed parameters.

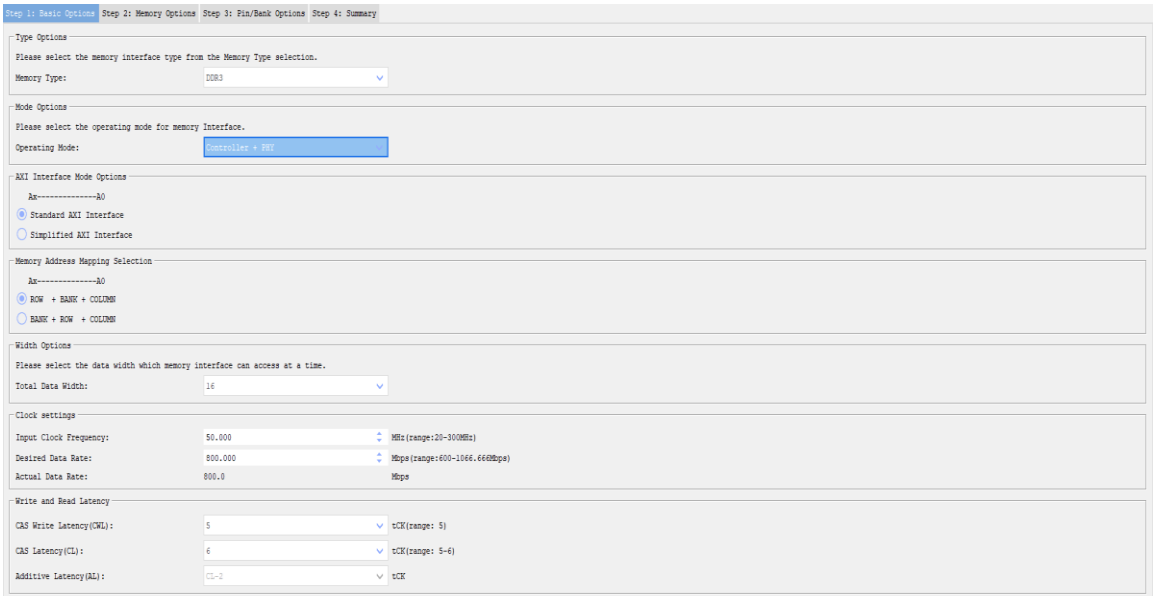

Figure 2-6 Basic Options Page

Table 2-2 Descriptions of Configuration Parameters on the Basic Options Page

Option Domain	**Option Name/Parameter Name**	**Parameter Description**	**Default Value**
Type Options	Memory Type	The type of SDRAM used, the currently optional type is: DDR3.	DDR3
Mode Options	Operating Mode	HMIC_S operating mode selection. The available operating modes are: 1) Controller +PHY 2) PHY Only; If Controller +PHY is selected, the generated IP code includes both the Controller and PHY; If PHY Only is selected, the generated IP code will only include the PHY part.	PHY Only
AXI Interface Mode Options[1]	Standard AXI Interface	Select the Controller AXI interface as Standard AXI4 mode;	Selected
	Simplified AXI interface	Select the Controller AXI interface as Simplified AXI4 mode;	Cleared
Memory Address Mapping Selection[2]	ROW + BANK + COLUMN	The read/write address mapping method for the Controller AXI interface is selected as: "ROW + BANK + COLUMN", please refer to Figure 2-16.	Selected
	BANK + ROW + COLUMN	The read/write address mapping method for the Controller AXI interface is selected as: "BANK + ROW + COLUMN", please refer to Figure 2-17.	Cleared
Width Options	Total Data Width[3]	The total DQ width of the off-chip SDRAM connected to HMIC_S. Supported total widths include: 72 64 56 48 40 32 24 16 8	16

1 This option is only displayed when the Operating Mode is "Controller + PHY".

2 This option is only displayed when the Operating Mode is "Controller + PHY".

3 The PG2L100H in FBG484 and MBG324 packages, as well as all PG2L25H and PG2L50H packages, only support a width of up to 32.

Option Domain	Option Name/Parameter Name	Parameter Description	Default Value
Clock Settings	Input Clock Frequency	The input clock of HMIC_S, in MHz.	50
	Desired Data Rate[4]	Desired data rate. The maximum speed supported by DDR3 is 1066Mbps. The maximum speed supported by DDR2 is 800Mbps. The maximum speed supported by LPDDR is 400Mbps.	800Mbps
	Actual Data Rate	The actual data rate, which should be as close to the desired rate as possible.	-
Write and Read Latency	CAS Write Latency(CWL)	CAS Write Latency configuration, in tCK. (Only DDR3 is displayed)	5
	CAS Latency(CL)	CAS Latency configuration, in tCK. (Only DDR3 is displayed)	6
	Additive Latency(AL)	Additive Latency configuration, in tCK. (Only DDR3 is displayed)	CL-2

Note: "-" indicates that there is no default value for this parameter in the IP configuration interface.

2.3.1.2.2 Step 2: Memory Options

Memory Options is the configuration page for Memory parameters, as shown in Figure 2-7. For detailed parameters, please refer to Table 2-3.

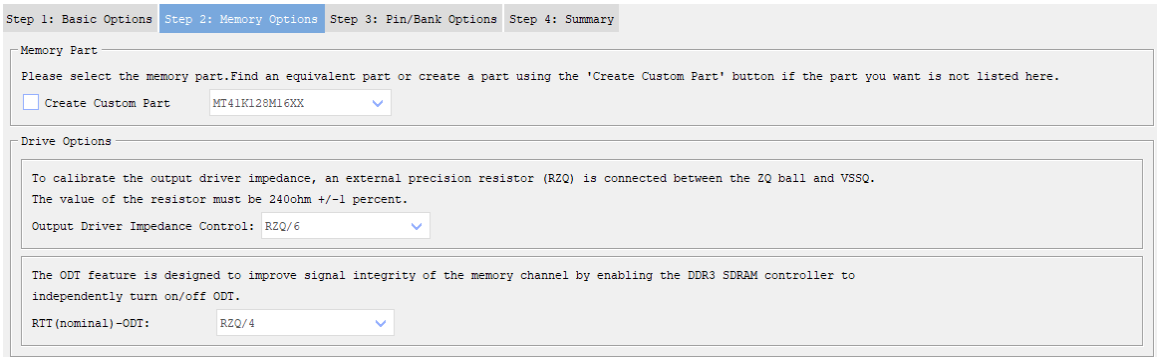

Figure 2-7 Memory Options Page

4 The default value for this parameter is related to the "Memory Type," DDR3: 800Mbps; DDR2: 600Mbps; LPDDR: 200Mbps.

Table 2-3 Descriptions of Configuration Parameters on the Memory Options Page

Option Domain	Option Name/Parameter Name	Parameter Description	Default Value
Memory Part	SDRAM device model[5]	The models supported by DDR3 include: MT41K128M8XX MT41K64M16XX MT41K256M8XX MT41K128M16XX MT41K512M8XX MT41K256M16XX	MT41K256 M16XX
	Create Custom Part	Custom new SDRAM type enable selection. If the device models supported by the IP do not meet the requirements, users can check this option and customize a new SDRAM type based on the reference model in the Custom Memory Part option box below.	Cleared
Custom Memory Part[6]	Select Base Part[7]	Custom SDRAM reference model.	MT41K256 M16XX
	Timing Parameters[8]	Custom SDRAM Timing parameters, including: trfc, tras, trp, trcd, twr, trefi, trtp, twtr.	trfc: 160 tras: 36 trp: 13.5 trcd: 13.5 twr: 15 trefi: 7.8 trtp: 7.5 twtr: 7.5
	Row Address[9]	Row address.	14
	Column Address[10]	Column address.	10
	Bank Address[11]	Bank address.	3

5 The default value for this parameter is related to the "Memory Type," DDR3: MT41K128M16XX; DDR2: MT47H128M16XX-25E; LPDDR: MT46H128M16XXXX-5L-IT.

6 This option box is displayed when the "Create Custom Part" option is selected, and it is used to customize a new SDRAM type.

7 The default value for this parameter is related to the "Memory Type," DDR3: MT41K128M16XX; DDR2: MT47H128M16XX-25E; LPDDR: MT46H128M16XXXX-5L-IT.

8 The default value for this parameter is related to the "Memory Type," DDR3: 1) trfc: 160, 2) tras: 36, 3) trp: 13.5, 4)trcd: 13.5, 5) twr: 15, 6) trefi: 7.8, 7) trtp: 7.5, 8) twtr: 7.5; DDR2: 1) trfc: 197.5, 2)tras: 40, 3) trp: 12.5, trcd: 12.5, 5)twr: 15, 6) trefi: 7.8, 7) trtp: 7.5, 8) twtr: 7.5; LPDDR: 1)trfc: 72, 2) tras: 38.4, 3) trp: 14.4, 4)trcd: 14.4, 5)twr: 14.4, 6)trefi: 7.8, 7) twtr: 2.

9 The default value for this parameter is related to the "Memory Type", DDR3: 14; DDR2: 14; LPDDR: 14.

10 The default value for this parameter is related to the "Memory Type", DDR3: 10; DDR2: 10; LPDDR: 11.

11 The default value for this parameter is related to the "Memory Type," DDR3: 3; DDR2: 3; LPDDR: 2.

Option Domain	Option Name/Parameter Name	Parameter Description	Default Value
Drive Options[12]	Output Driver Impedance Control	Drive strength options supported by DDR3. For detailed information, please refer to "***JESD79-3D, DDR3 SDRAM Standard***".	RZQ/6
	RTT(nominal)-ODT		RZQ/4

2.3.1.2.3 Step 3: Pin/Bank Options

Pin/Bank Options is the configuration page for interface parameters, as shown in Figure 2-8. Please refer to Table 2-4 for detailed parameter descriptions.

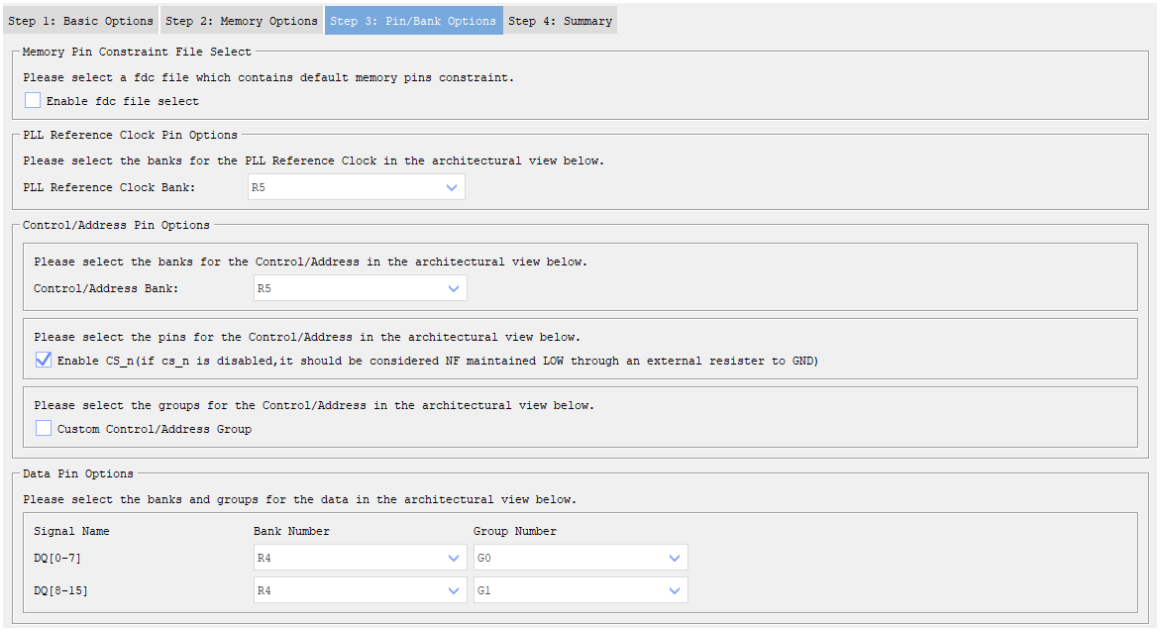

Figure 2-8 Pin/Bank Options Page

12 The option parameters related to drive strength and their default configurations are associated with the "Memory Type". The drive strength options for different types of SDRAM and their default configurations are described as follows:

DDR3: Output Driver Impedance Control (default configuration: RZQ/6), RTT(nominal)-ODT (default configuration: RZQ/4);

DDR2: Output Driver Strength (default configuration: Full strength), RTT(nominal)-ODT (default configuration: 75ohms);

LPDDR: Driver Strength (default configuration: Full strength).

Attention:

All configuration items in "Step 3: Pin/Bank Options" must be configured according to the actual pin assignments on the circuit board. After generating the IP, the DQ pins,reset pins and status pins should be constrained according to actual pin assignments on the cuicuit board, otherwise errors may occur during the flow process.

Table 2-4 Description of Configuration Parameters on Pin/Bank Options Page

Option Domain	**Option Name/Parameter Name**	**Parameter Description**	**Default Value**
Memory Pin Constraint File Select	Enable fdc file select	Custom .fdc files can be imported. When enabled, users can input the path to their fdc file in the text box. It will automatically read the Memory interface constraints from the fdc file and configure the Control/Address and Data Pins in the UI. Notes: The signal names of the Memory interface in the user's fdc must match those in the Example Design.	Disabled
PLL Reference Clock Pin Options	PLL Reference Bank[13]	The bank where the PLL reference clock is located.	L5
Control/ Address Pin Options	Control/Address Bank[13]	The bank where the Memory interface's control and address lines are located.	L5
	Enable CS_n	mem_cs_n signal enable selection.	Enabled
	Custom Control/Address Group	User-defined control and address bus grouping enable selection. Selected: User defines the grouping and pin constraints for each PAD; Cleared: default grouping.	Disabled
	Control and address signals	This option box is displayed when the "Custom Control/Address Group" option is selected, and it is used to select the Group and Pin where the control and address signals corresponding PADs are located.	-

13 The default value for this parameter is related to the selected device and package: PG2L25H: 1) MBG325: L5; PG2L50H: 1) MBG324: L5, 2) FBG484: L4; PG2L100H: 1) FBG484: R5, 2) FBG676: L5, 3) MBG324: R5; PG2T390H: 1) FFBG900: L3, 2) FFBG676: L5; PG2L200H: 1) FBB484: R5, 2) FBB676: L5, 3) FFBG1156: R4; PG2L100HX: 1) FBG484: R5, 2) FBG676: L5, 3) MBG324: R5; PG2K400: 1) FFBG900: L8, 2) FFBG676: L6; PG2T70H: 1) FBB484: L5, 2) FBB676: L5.

Option Domain	Option Name/Parameter Name	Parameter Description	Default Value
Data Pin Options	Bank Number[14]	Select the bank where DQ[15] is located.	L6
	Group Number[16]	Select the group where DQ[15] is located.	G1

Note: "-" indicates that there is no default value for this parameter in the IP configuration interface.

Attention:

For the "Custom Control/Address Group" option, it is recommended to use custom grouping due to different PCB traces in actual use.

When checking the "Custom Control/Address Group" option for custom pin configuration, ensure that signals are not constrained to the same pin. If pins are constrained to the same location, the UI will highlight the corresponding Pin Number in red.

2.3.1.2.4 Step 4: Summary

The Summary page is used to print the current configuration information without configuration required, as shown in Figure 2-9.

14 The default configuration value of this parameter depends on the selected device and package. For example, for DQ[0-7], PG2L25H: 1) MBG325: L4; PG2L50H: 1) MBG324: L4, 2) FBG484: L3; PG2L100H: 1) FBG484: R4, 2) FBG676: L6, 3) MBG324: R4; PG2T390H: 1) FFBG900: L2, 2) FFBG676: L6; PG2L200H: 1) FBB484: R4, 2) FBB676: L6, 3) FFBG1156: R3; PG2L100HX: 1) FBG484: R4, 2) FBG676: L6, 3) MBG324: R4; PG2K400: 1) FFBG900: L9, 2) FFBG676: L7; PG2T70H: 1) FBB484: L4, 2) FBB676: L6.

15 DQ[8-15] is displayed when the data width is greater than 8, DQ[16-23] is displayed when the data width is greater than 16, DQ[24-31] is displayed when the data width is greater than 24, DQ[32-39] is displayed when the data width is greater than 32, DQ[40-47] is displayed when the data width is greater than 40, DQ[48-55] is displayed when the data width is greater than 48, DQ[56-63] is displayed when the data width is greater than 56, DQ[64-71] is displayed when the data width is greater than 64.

16 The default configuration value of this parameter depends on the selected device and package. For example, for DQ[0-7], PG2L25H: 1) MBG325: G1; PG2L50H: 1) MBG324: G0, 2) FBG484: G2; PG2L100H: 1) FBG484: G0, 2) FBG676: G1, 3) MBG324: G2; PG2T 390H: 1) FFBG900: G0, 2) FFBG676: G3; PG2L200H: 1) FBB484: G0, 2) FBB676: G1, 3) FFBG1156: G1; PG2L100HX: 1) FBG484: G0, 2) FBG676: G1, 3) MBG324: G2; PG2K400: 1) FFBG900: G2, 2) FFBG676: G0; PG2T70H: 1) FBB484: G0, 2) FBB676: G3.

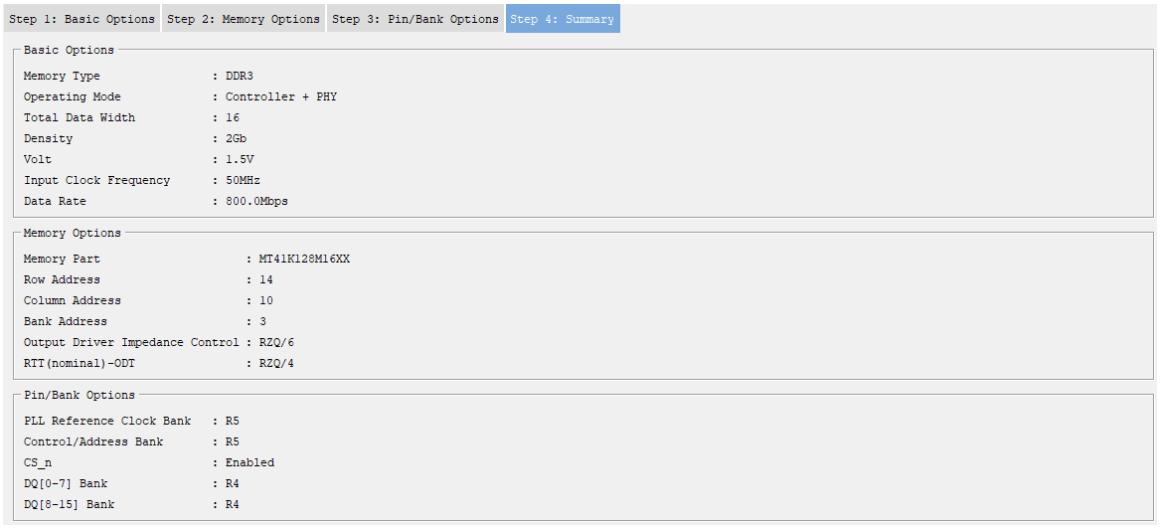

Figure 2-9 Summary Page

2.3.1.3 Generating IP

After completing the parameter configuration, click the <Generate> button in the top left corner to generate IP and the HMIC_S IP code corresponding to the user-specific settings. The information report interface for IP generation is shown in Figure 2-10.

Figure 2-10 HMIC_S IP Generation Report Interface

Attention:

The .pds and .fdc files generated with the IP are for reference only; please modify the pin constraints according to the actual pin connections when in use.

Upon successful IP generation, the files indicated in Table 2-5 will be output to the Project path specified in Figure 2-3.

Table 2-5 Output Files after IP Generation

Output File[17]	**Description**
$instname.v	The top-level .v file of the generated IP for Controller + PHY.
$instname_ddrphy_top.v	The top-level .v file of the generated IP for PHY.
$instname.idf	The Configuration file of the generated IP.
/rtl/*	The RTL code file of the generated IP.
/example_design/*	The Test Bench and the corresponding Memory Simulation Model files used by the Example Design of the generated IP.
/pnr/*	The project files .pds and pin constraint files .fdc for the Example Design of the generated IP.
/sim/*	The simulation directory for the generated IP. sim.tcl is a ModelSim simulation script, makefile is a VCS simulation script, and sim_file_list.f is a list of simulation files.
/sim_lib/*	The directory of the encryption files for the IP.
/rev_1	The default output path for synthesis reports. (This folder is generated only after specifying the synthesis tool)
readme.txt	The readme file describes the structure of the generation directory after the IP is generated.

2.3.2 Constraint Configuration

For the specific configuration method of constraint files, please refer to the relevant help documents in the PDS installation path: "***User_Constraint_Editor_User_Guide***", "***Physical_Constraint_Editor_User_Guide***", ***and*** "***Route_Constraint_Editor_User_Guide***".

2.3.3 Simulation Runs

The simulation of HMIC_S IP is based on the Test Bench of the Example Design. For detailed information about Example Design, please refer to "2.4 Example Design".

For more details about the PDS simulation functions and third-party simulation tools, please consult the related help documents in the PDS installation path: "***Pango_Design_Suite_User Guide***", "***Simulation_User_Guide***".

17 <$instname> is the instantiation name entered by the user; "*" is a wildcard character used to replace files of the same type.

2.3.3.1 ModelSim Simulation

Simulation operation steps: Open cmd.exe, switch the current directory to "/sim/modelsim" under the IP generation directory in the command line, run vsim (vsim-64 is used on the server), and open the ModelSim simulation software, as shown in Figure 2-11.

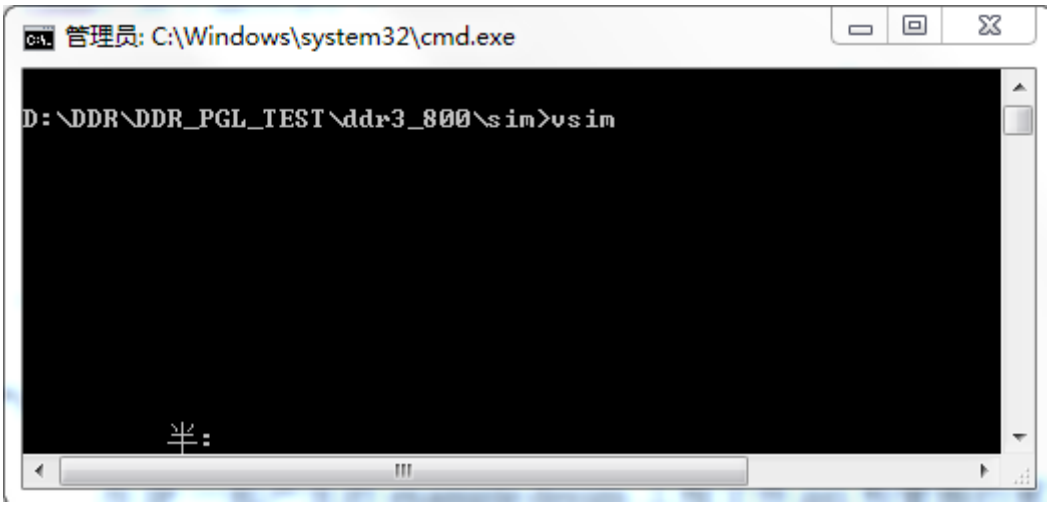

Figure 2-11 Open ModelSim Instruction

Run the ctrl_phy_sim.tcl script in the ModelSim simulation software to perform simulation, as shown in Figure 2-12.

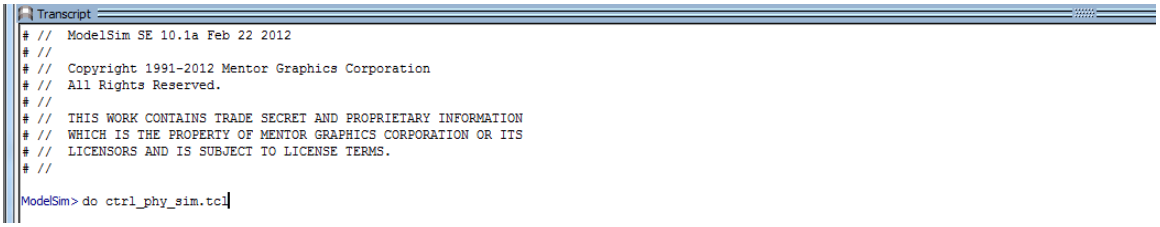

Figure 2-12 ModelSim Simulation Execution Script

2.3.3.2 VCS simulation

Enter the "sim/vcs" directory in Terminal, type "make" and press "Enter" to start the simulation, as shown in Figure 2-13.

Figure 2-13 VCS Simulation Script Execution

2.3.4 Synthesis and Placement/Routing

The specific usage of PDS synthesis tools and placement/routing tools can be found in the help documents within the PDS installation path.

Attention:

Example Design project file .pds and pin constraint file .fdc generated with the IP are located in the "/pnr/" directory, and physical constraints need to be modified according to the actual devices and PCB trace routing. For details, please refer to "2.8 Descriptions and Considerations".

2.3.5 Resources Utilization

Table 2-6 Typical Resource Utilization Values for HMIC_S IP Based on Applicable Devices

Device	Configuration Mode	IP Operating Mode	Typical Resource Utilization Values				
			LUT	FF	GPLL	PPLL	USCM
PG2L25H	DDR3 x32	Controller + PHY	4208	4094	2	2	2
PG2L50H	DDR3 x32	Controller + PHY	4200	4094	2	2	2
PG2L100H	DDR3 x32	Controller + PHY	4711	4179	2	2	2
	DDR3 x64	Controller + PHY	6885	5794	2	3	2
	DDR3 x32	Controller + PHY, Standard AXI4	7042	9214	2	2	2
PG2L100HX	DDR3 x32	Controller + PHY	4290	4148	2	2	2
	DDR3 x64	Controller + PHY	6479	5732	2	3	2
PG2L200H	DDR3 x32	Controller + PHY	4209	4094	2	2	2
	DDR3 x64	Controller + PHY	6274	5676	2	3	2
PG2T390H	DDR3 x16	Controller + PHY	3528	3373	2	2	2
	DDR3 x32	Controller + PHY	4703	4183	2	2	2
PG2K400	DDR3 x32	Controller + PHY	4260	4150	2	2	2
	DDR3 x64	Controller + PHY	6356	5734	2	3	2

Device	Configuration Mode	IP Operating Mode	Typical Resource Utilization Values				
			LUT	FF	GPLL	PPLL	USCM
PG2T70H	DDR3 x32	Controller + PHY	4171	4148	2	2	2
	DDR3 x64	Controller + PHY	6235	5730	2	3	2
PG2T390HX	DDR3 x32	Controller + PHY	4262	4094	2	2	2
	DDR3 x64	Controller + PHY	6343	5676	2	3	2
PG2T160H	DDR3x32	Controller + PHY, Simplified AXI4	4288	4233	2	2	2
	DDR3x32	Controller + PHY, Standard AXI4	6360	8578	2	2	2

2.4 Example Design

This section mainly introduces the Example Design scheme based on HMIC_S IP (Controller + PHY mode). In this scheme, the user logic acts as AXI Master with the HMIC_S IP as AXI Slave. The user logic writes data through the Write channel of the AXI interface and receives data through the Read channel for data comparison. If there is a data error, the Error LED will light up.

2.4.1 Design Block Diagram

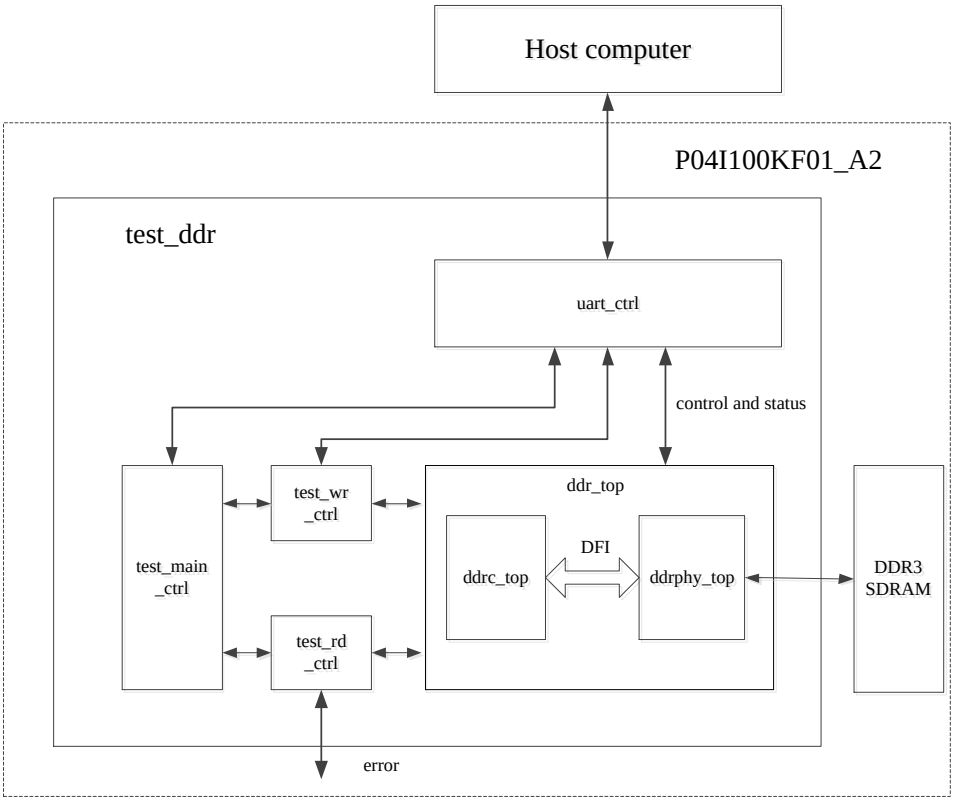

Figure 2-14 Example Design System Block Diagram

The system block diagram of the Example Design is shown in Figure 2-14, where test_main_ctrl is the control module for AXI read and write instructions, test_wr_ctrl is the control module for AXI write instructions and data writing, test_rd_ctrl is the control module for AXI read instructions and data reading, and uart_ctrl is the serial port conversion module for easy control and internal state reading during debugging.

2.4.2 Test Method

In the Example Design, the user logic performs read and write operations on HMIC_S IP and checks the readback data. The detailed test process is as shown in Figure 2-15.

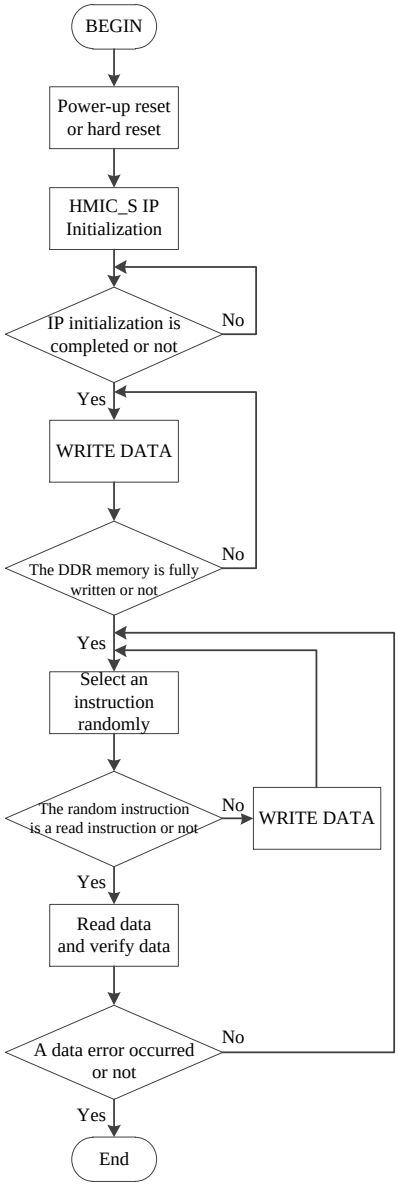

Figure 2-15 Example Design Test Flowchart

After the system is powered up or a hard reset is initiated, the HMIC_S IP begins initialization. Once initialization is completed (indicated by the ddrc_init_done signal going high), the test_main_ctrl module controls the test_wr_ctrl module to generate write instructions and write data to initialize the DDR chips. After write full, test_main_ctrl starts random read and write operations, and test_rd_ctrl checks the readback data to determine if there are any errors.

Attention:

Do not directly use the Example Design generated by the IP for on-board testing flow. Constrain pins according to the actual pin connections of the board, then proceed with Flow on-board testing.

2.5 Descriptions of IP Interfaces

This section provides the HMIC_S IP related interface instructions and timing descriptions.

2.5.1 Controller Interface Description

2.5.1.1 Global Interface

Table 2-7 Global Interface

Port	I/O	Bit width	Valid Values	Description
clk	I	1	-	External clock input.
rst_n	I	1	Low	External reset input.
phy_init_done	I	1	High	Initialization completion flag for ddrphy: 1'b1: ddrphy initialization is complete; 1'b0: DDRPHY initialization is incomplete;
ddr_init_done	O	1	High	Initialization completion flag for the IP: 1'b1: ddr IP initialization is complete; 1'b0: ddr IP initialization is incomplete, and external operations on the ddr IP are invalid.

Note: "-" indicates that the parameter does not exist.

2.5.1.2 Simplified AXI4 interface

2.5.1.2.1 Write address channel

Table 2-8 Write Address Channel

Port	I/O	Bit width	Valid Values	Description
axi_awaddr	I	CTRL_ADDR_WIDTH	-	AXI write address.
axi_awuser_ap	I	1	High	AXI write with auto precharge.
axi_awuser_id	I	4	-	AXI write address ID.
axi_awlen	I	4	-	AXI burst length for writing.
axi_awready	O	1	High	AXI write address ready.
axi_awvalid	I	1	High	AXI write address valid.

Note: "-" indicates that the parameter does not exist.

2.5.1.2.2 Read address channel

Table 2-9 Read Address Channel

Port	I/O	Bit width	Valid Values	Description
axi_araddr	I	CTRL_ADDR_WIDTH	-	AXI read address.
axi_aruser_ap	I	1	High	AXI read with auto precharge.
axi_aruser_id	I	4	-	AXI read address ID.
axi_arlen	I	4	-	AXI burst length for reading.
axi_arready	O	1	High	AXI read address ready.
axi_arvalid	I	1	High	AXI read address valid

Note: "-" indicates that the parameter does not exist.

2.5.1.2.3 Write data channel

Table 2-10 Write Data Channel

Port	I/O	Bit width	Valid Values	Description
axi_wdata	I	DQ_WIDTH*8	-	AXI write data.
axi_wstrb	I	DQ_WIDTH*8/8	High	AXI write data strobes.
axi_wready	O	1	High	AXI write data ready.
axi_wusero_id	O	4	-	AXI write data ID.
axi_wusero_last	O	1	High	AXI write data last.

Note: "-" indicates that the parameter does not exist.

2.5.1.2.4 Read data channel

Table 2-11 Read Data Channel

Port	I/O	Bit width	Valid Values	Description
axi_rid	O	4	-	AXI read data ID.
axi_rlast	O	1	High	AXI read data last signal.
axi_rvalid	O	1	High	AXI read data valid.
axi_rdata	O	DQ_WIDTH*8	-	AXI read data.

Note: "-" indicates that the parameter does not exist.

2.5.1.3 Standard AXI4 Interface

2.5.1.3.1 Write address channel

Table 2-12 Write Address Channel

Port	I/O	Bit width	Valid Values	Description
axi_awaddr	I	AXI_ADDR_WIDTH	-	AXI write address.
axi_awid	I	8	-	AXI write address ID.
axi_awsize	I	3	-	AXI write burst size.
axi_awburst	I	2	-	AXI write burst type.
axi_awlen	I	8	-	AXI burst length for writing.
axi_awready	O	1	High	AXI write address ready.
axi_awvalid	I	1	High	AXI write address valid.

Note: "-" indicates that the parameter does not exist.

2.5.1.3.2 Read address channel

Table 2-13 Read Address Channel

Port	I/O	Bit width	Valid Values	Description
axi_araddr	I	AXI_ADDR_WIDTH	-	AXI read address.
axi_arid	I	8	-	AXI read address ID.
axi_arsize	I	3	-	AXI read burst size.
axi_arburst	I	2	-	AXI read burst type.
axi_arlen	I	8	-	AXI burst length for reading.
axi_arready	O	1	High	AXI read address ready.
axi_arvalid	I	1	High	AXI read address valid.

Note: "-" indicates that the parameter does not exist.

2.5.1.3.3 Write data channel

Table 2-14 Write Data Channel

Port	I/O	Bit width	Valid Values	Description
axi_wdata	I	AXI_DATA_WIDTH*8	-	AXI write data.
axi_wstrb	I	AXI_DATA_WIDTH*8/8	High	AXI write data strobes.
axi_wvalid	I	1		AXI write data valid.
axi_wready	O	1	High	AXI write data ready.
axi_bready	I	1	High	AXI write response ready.
axi_bid	O	8	-	AXI write data ID.
axi_bresp	O	2	-	AXI write response.
axi_bvalid	O	1		AXI write response valid.
axi_wlast	I	1	High	AXI write data last.

Note: "-" indicates that the parameter does not exist.

2.5.1.3.4 Read data channel

Table 2-15 Read Data Channel

Port	I/O	Bit width	Valid Values	Description
axi_rid	O	8	-	AXI read data ID.
axi_rlast	O	1	High	AXI read data last signal.
axi_rvalid	O	1	High	AXI read data valid.
axi_rdata	O	AXI_DATA_WIDTH*8	-	AXI read data.
axi_rready	I	1		AXI read data ready. Note: axi_rready needs to be pulled high whenever axi_rvalid is active.
axi_rresp	O	2		AXI read response.

Note: "-" indicates that the parameter does not exist.

2.5.1.4 Config port

Table 2-16 Config Interface

Port	I/O	Bit width	Valid Values	Description
apb_clk	I	1	High	APB clock.
apb_rst_n	I	1	Low	APB reset.
apb_sel	I	1	High	APB Select.
apb_enable	I	1	High	APB port enable.
apb_addr	I	8	-	APB address bus.
apb_write	I	1	High	APB read/write direction, high level for write, active low for read.

Port	I/O	Bit width	Valid Values	Description
apb_ready	O	1	High	APB port ready.
apb_wdata	I	16	-	APB write data.
apb_rdata	O	16	-	APB read data.

Note: "-" indicates that the parameter does not exist.

2.5.1.5 DFI interface

Table 2-17 DFI Interface

Port	I/O	Bit width	Valid Values	Description
dfi_address	O	4*ROW_ADDR_WIDTH	-	DFI address bus.
dfi_bank	O	4*BADDR_WIDTH	-	DFI bank address bus.
dfi_reset_n	O	4	-	DFI chip reset.
dfi_cs_n	O	4	-	DFI chip select.
dfi_ras_n	O	4	-	DFI row address strobe bus.
dfi_cas_n	O	4	-	DFI column address strobe.
dfi_we_n	O	4	-	DFI write enable signal.
dfi_cke	O	4	-	DFI clock enable.
dfi_odt	O	4	-	DFI on-die termination control bus.
dfi_rddata_valid	I	1	High	Read data valid on DFI interface.
dfi_rddata	I	DQ_WIDTH*8	-	DFI interface data.
dfi_wrdata_en	O	4	High	DFI write enable signal.
dfi_wrdata	O	DQ_WIDTH*8	-	DFI write data.
dfi_wrdata_mask	O	DQ_WIDTH*8/8	High	DFI Write data byte mask.
dfi_init_complete	I	1	High	PHY has completed training operations and is now in a normal state.
dfi_error	I	1	High	PHY training error indication.
dfi_phyupd_req	I	1	-	PHY requests an update.
dfi_phyupd_ack	O	1	-	Feedback signal of dfi_phy_req, allowing PHY to update.

Note: "-" indicates that the parameter does not exist.

2.5.2 Controller Interface Timing Description

The Simplified AXI4 interface uses a trimmed version of the AXI4 protocol, while the Standard AXI4 interface uses the standard AXI4 protocol. The Config interface uses the APB protocol.

2.5.2.1 Simplified AXI4 interface

For differences between the Simplified AXI4 interface as defined in this design and the standard AXI4 protocol, please refer to Table 2-18 and Table 2-19.

Table 2-18 Differences between Simplified AXI4 and Standard AXI4

Channel	**Differences**
Write address channel	Preserve AWID, AWADDR, AWLEN, AWUSER, AWVALID, and AWREADY ports; remove AWSIZE, AWBURST, AWLOCK, AWCACHE, AWPROT, AWQOS, and AWREGION ports.
Write data channel	Preserve WID, WDATA, WSTRB, WLAST, WREADY ports; remove WUSER and WVALID ports.
Write responsechannel	N/A.
Read address channel	Preserve ARID, ARADDR, ARLEN, ARUSER, ARVALID, and ARREADY ports; remove ARSIZE, ARBURST, ARLOCK, ARCACHE, ARPROT, ARQOS, and ARREGION ports.
Read data channel	Preserve RID, RDATA, RLAST, and RVALID; remove RRESP and RUSER ports.
Low-power interface	N/A.
Clock	The clock for AXI and MC is the same.

Table 2-19 Detailed Differences between Simplified AXI4 and Standard AXI4

Interface Signal Classification	**Standard AXI4**	**Simplified AXI4**	**Signal Source**	**Description**	**Compatibility Method**
Global signals	ACLK	clk	Clock Source	Global clock signal	PHY core division clock
	ARESETn	rst_n	Reset Source	Global reset signal, active-low	MC reset clock
Write address channel signals	AWID[18]	axi_awuser_id	Master	Write address ID	Assign a fixed value when not in use
	AWADDR[18]	axi_awaddr	Master	Write address	Refer to the footnote
	AWLEN[18]	axi_awlen	Master	Write burst length	Default burst size is 8
	AWSIZE[19]	N/A	Master	Write burst size	Refer to the footnote
	AWBURST[19]	N/A	Master	Burst type	Refer to the footnote

18 For interface timing, please refer to "2.5.2.1.1 Write address channel timing", "2.5.2.1.2 Read address channel timing", "2.5.2.1.3 Write data channel timing", and "2.5.2.1.4 Read data channel timing".

19 On the basis of the DDR3 IP instantiation module, a package module layer is provided, reserving input interfaces not connected to the controller as input signals.

Interface Signal Classification	Standard AXI4	Simplified AXI4	Signal Source	Description	Compatibility Method
	AWLOCK[19]	N/A	Master	Lock type	Refer to the footnote
	AWCACHE[19]	N/A	Master	Cache type	Refer to the footnote
	AWPROT[19]	N/A	Master	Protection type	Refer to the footnote
	AWQOS[19]	N/A	Master	Write QOS identifier	Refer to the footnote
	AWREGION[19]	N/A	Master	Write domain identifier	Refer to the footnote
	AWUSER[18]	axi_awuser_ap	Master	User-defined, write with auto precharge	Refer to the footnote
	AWVALID[18]	axi_awvalid	Master	Write address valid	Refer to the footnote
	AWREADY[18]	axi_awready	Slave	Write address ready	Refer to the footnote
Write data channel signals	WID[18]	axi_wusero_id	Master	Write ID	Assign a fixed value when not in use
	WDATA[18]	axi_wdata	Master	WRITE DATA	Refer to the footnote
	WSTRB[18]	axi_wstrb	Master	Write strobe	Refer to the footnote
	WLAST[18]	axi_wusero_last	Master	Write last flag	Refer to the footnote
	WUSER[19]	N/A	Master	User defined	Refer to the footnote
	WVALID[19]	N/A	Master	Write valid	Refer to the footnote
	WREADY[18]	axi_wready	Slave	Write ready	Refer to the footnote
Write response channel signals	BID[20,21]	N/A	Slave	Response ID	Refer to the footnote
	BRESP[20,21]	N/A	Slave	Write response	Refer to the footnote
	BUSER[20,21]	N/A	Slave	User defined	Refer to the footnote
	BVALID[20,21]	N/A	Slave	Write response valid	Refer to the footnote
	BREADY[19,21]	N/A	Master	Write response ready	Refer to the footnote

20 On the basis of the DDR3 IP instantiation module, a package module layer is provided, reserving output interfaces that can be assigned a fixed value as output signals.

21 For MC, as there is no write response channel provided internally, it is recommended that the master device ignores the write response channel function.

Interface Signal Classification	Standard AXI4	Simplified AXI4	Signal Source	Description	Compatibility Method
Read address channel signals	ARID[18]	axi_aruser_id	Master	Read ID	Assign a fixed value when not in use
	ARADDR[18]	axi_araddr	Master	Read data address	Refer to the footnote
	ARLEN[18]	axi_arlen	Master	Burst length	Refer to the footnote
	ARSIZE[19]	N/A	Master	Burst size	Refer to the footnote
	ARBURST[19]	N/A	Master	Burst type	Refer to the footnote
	ARLOCK[19]	N/A	Master	Lock type	Refer to the footnote
	ARCACHE[19]	N/A	Master	Cache type	Refer to the footnote
	ARPROT[19]	N/A	Master	Inclusion type	Refer to the footnote
	ARQOS[19]	N/A	Master	Read address QOS identifier	Refer to the footnote
	ARREGION[19]	N/A	Master	Read address domain identifier	Refer to the footnote
	ARUSER[18]	axi_aruser_ap	Master	User-defined, read with auto precharge	Refer to the footnote
	ARVALID[18]	axi_arvalid	Master	Read address valid	Refer to the footnote
	ARREADY[18]	axi_arready	Slave	Read address ready	Refer to the footnote
Read data channel signals	RID[18]	axi_rid	Slave	Read ID	Refer to the footnote
	RDATA[18]	axi_rdata	Slave	Read data	Refer to the footnote
	RRESP[20]	N/A	Slave	Read response	Refer to the footnote
	RLAST[18]	axi_rlast	Slave	Read last data	Refer to the footnote
	RUSER[20]	N/A	Slave	User defined	Refer to the footnote
	RVALID[18]	axi_rvalid	Master	Read valid	Refer to the footnote
	RREADY[19]	N/A	Master	Read ready	Refer to the footnote
Low-power interface signals	CSYSREQ[19]	N/A	Clock controller	System low power request	Refer to the footnote
	CSYSACK[19]	N/A	Peripheral devices	Low power request response	Refer to the footnote
	CACTIVE[19]	N/A	Peripheral devices	Clock activity	Refer to the footnote

The Simplified AXI4 interface comprises four channels, including write address channel, read address channel, write data channel, and read data channel, each operating independently. Generally, the address must be sent before transmitting and receiving data. Users can determine if the returned data matches the request using different ID numbers.

There are two types of correspondence between AXI4 interface addresses and Memory addresses, corresponding to the "Memory Address Mapping Selection" option area in the configuration interface. When "ROW+BANK+COLUMN" is selected, the correspondence between AXI4 interface addresses and Memory addresses is shown in Figure 2-16; when "BANK+ROW+COLUMN" is selected, the correspondence between AXI4 interface addresses and Memory addresses is as shown in Figure 2-17.

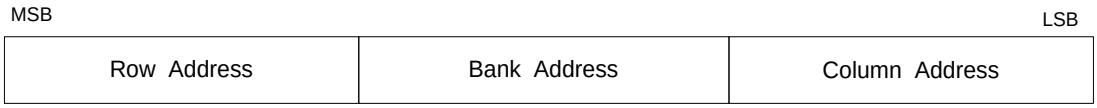

Figure 2-16 Memory Address 1 Mapped From AXI4 Interface Address

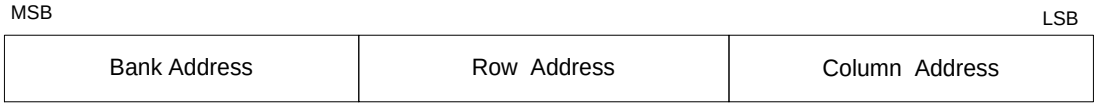

Figure 2-17 Memory Address 2 Mapped From AXI4 Interface Address

Descirption:

When users request is at the column boundary, column address rounding will occur. For related examples, please refer to "2.8.2 Column Address Rounding Examples".

2.5.2.1.1 Write address channel timing

Signals included in the write address channel: axi_awready, axi_awvalid, axi_awaddr, axi_awuser_ap, axi_awuser_id, axi_awlen. A typical timing is shown in Figure 2-18.

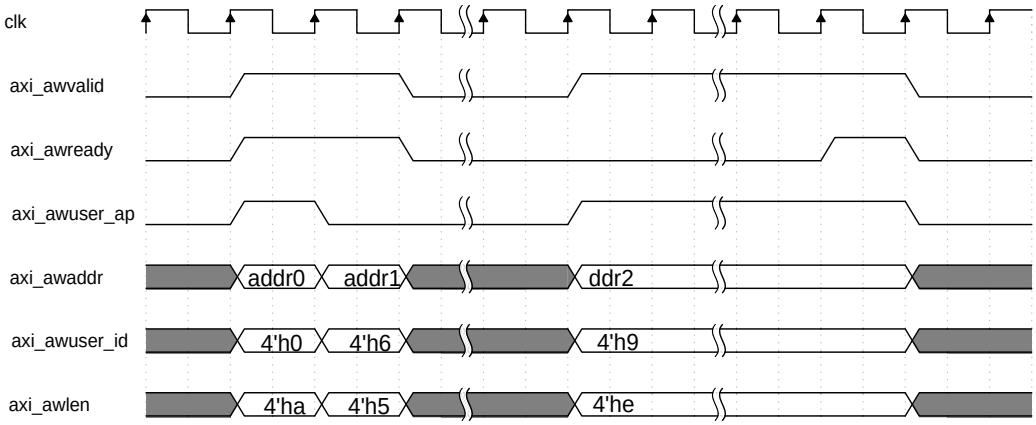

Figure 2-18 Typical Timing for Write Address

- The handshake is completed when axi_awready and axi_awvalid are both valid simultaneously.
- The packet length is controlled by axi_awlen, which is the value of axi_awlen plus one.
- Handshake process: starting from the rising edge of the clock when axi_awvalid is valid, axi_awaddr, axi_awuser_ap, axi_awuser_id, axi_awlen must remain unchanged until the handshake is completed and released.

2.5.2.1.2 Read address channel timing

Signals included in the read address channel: axi_arready, axi_arvalid, axi_araddr, axi_aruser_ap, axi_aruser_id, axi_arlen. A typical timing is shown in Figure 2-19.

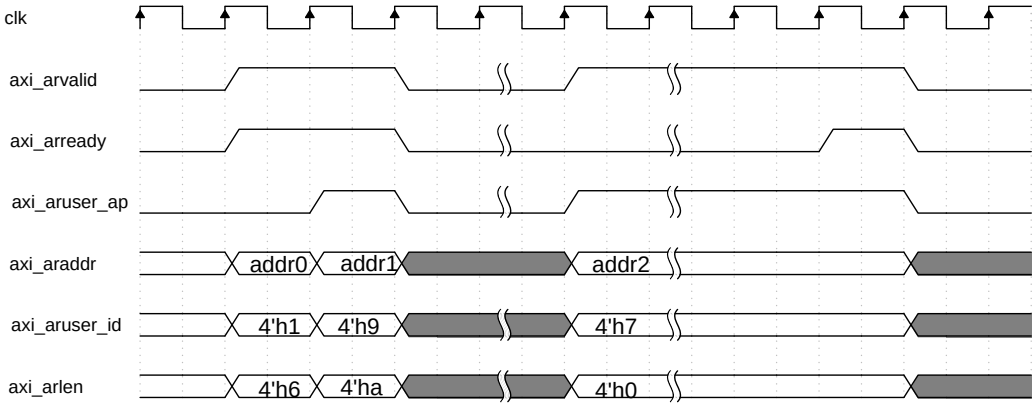

Figure 2-19 Typical Timing for Read Address

- The handshake is completed when axi_arready and axi_arvalid are both valid simultaneously.
- The packet length is controlled by axi_arlen, which is the value of axi_arlen plus one.
- Handshake process: starting from the rising edge of the clock when axi_arvalid is valid, axi_araddr, axi_aruser_ap, axi_aruser_id, axi_arlen must remain unchanged until the handshake is completed and released.

2.5.2.1.3 Write data channel timing

Signals included in the write data channel: axi_wready, axi_wusero_id, axi_wusero_last, axi_wdata, and axi_wstrb. A typical timing is shown in Figure 2-20.

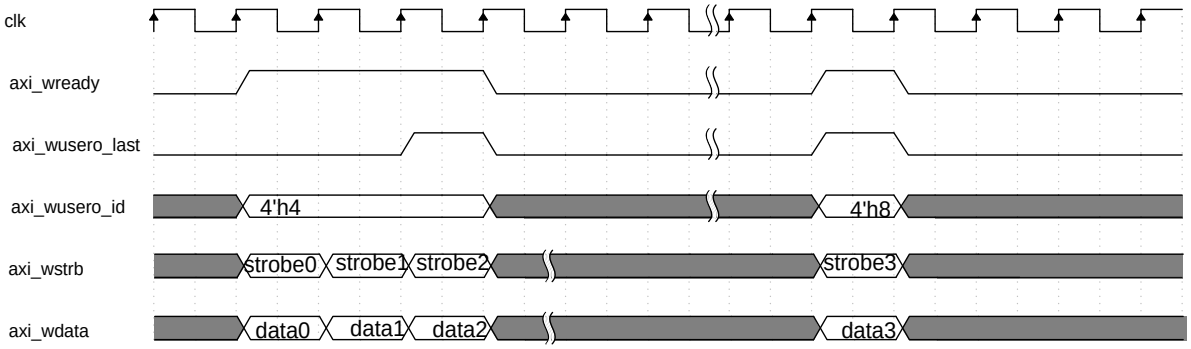

Figure 2-20 Typical Timing for Write Data

- The transmission is indicated as valid when axi_wready is valid.
- The end of a transmission is indicated when axi_wusero_last is valid.
- During transmission: axi_wready, axi_wusero_id, and axi_wusero_last are received synchronously, axi_wdata and axi_wstrb are transmitted synchronously.

2.5.2.1.4 Read data channel timing

Signals included in the read data channel: axi_rdata, axi_rid, axi_rlast, and axi_rvalid. A typical timing is shown in Figure 2-21.

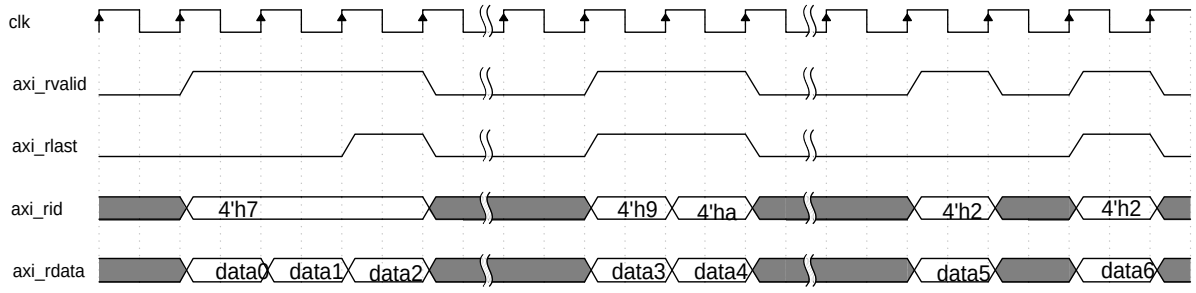

Figure 2-21 Typical Timing for Read Data

- The transmission is indicated as valid when axi_rvalid is valid.
- The end of a transmission is indicated when axi_rlast is valid.

2.5.2.2 Standard AXI4 Interface

2.5.2.2.1 Write address channel timing

Signals included in the write address channel: axi_awready, axi_awvalid, axi_awaddr, axi_awsize, axi_awburst, axi_awid, and axi_awlen. A typical timing is shown in Figure 2-22.

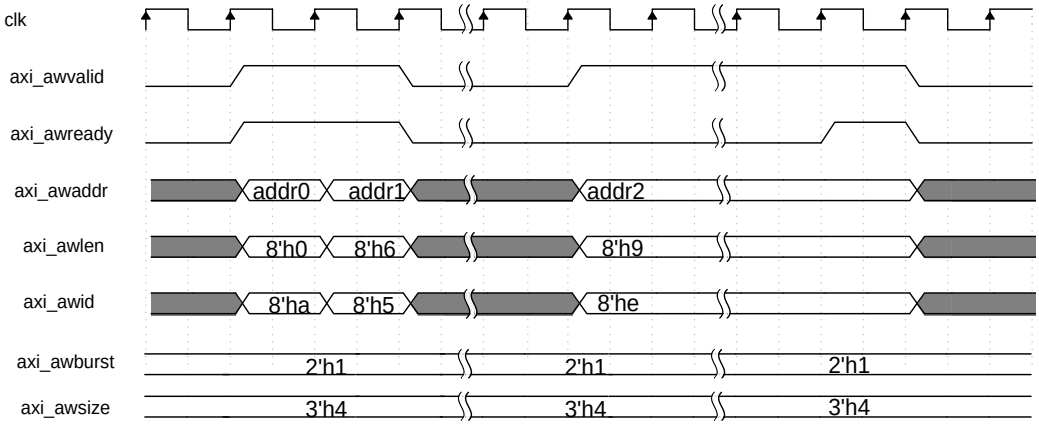

Figure 2-22 Typical Timing for Write Address

- The handshake is completed when axi_awready and axi_awvalid are both valid simultaneously.
- The packet length is controlled by axi_awlen, which is the value of axi_awlen plus one.
- Handshake process: starting from the rising edge of the clock when axi_awvalid is valid, axi_awaddr, axi_awid and axi_awlen must remain unchanged until the handshake is completed and released.

2.5.2.2.2 Read address channel timing

Signals included in the read address channel:axi_arready, axi_arvalid, axi_araddr, axi_arburst, axi_arsize, axi_arid and axi_arlen. A typical timing is shown in Figure 2-23.

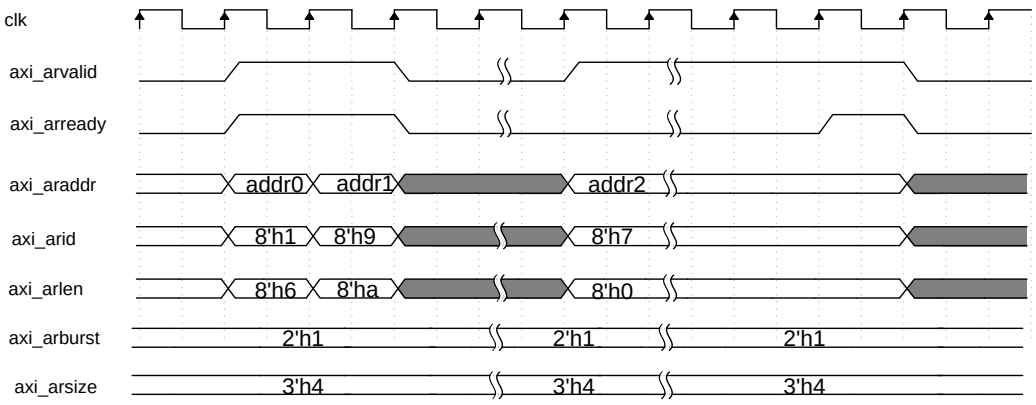

Figure 2-23 Typical Timing for Read Address

- The handshake is completed when axi_arready and axi_arvalid are both valid simultaneously.
- The packet length is controlled by axi_arlen, which is the value of axi_arlen plus one.
- Handshake process: starting from the rising edge of the clock when axi_arvalid is active, axi_araddr, axi_arid, and axi_arlen must remain unchanged until the handshake is completed and released.

2.5.2.2.3 Write data channel timing

Signals included in the write data channel: axi_wready, axi_wvalid, axi_bready, axi_bid, axi_bresp, axi_bvalid, axi_wlast, axi_wdata, and axi_wstrb. A typical timing is shown in Figure 2-24.

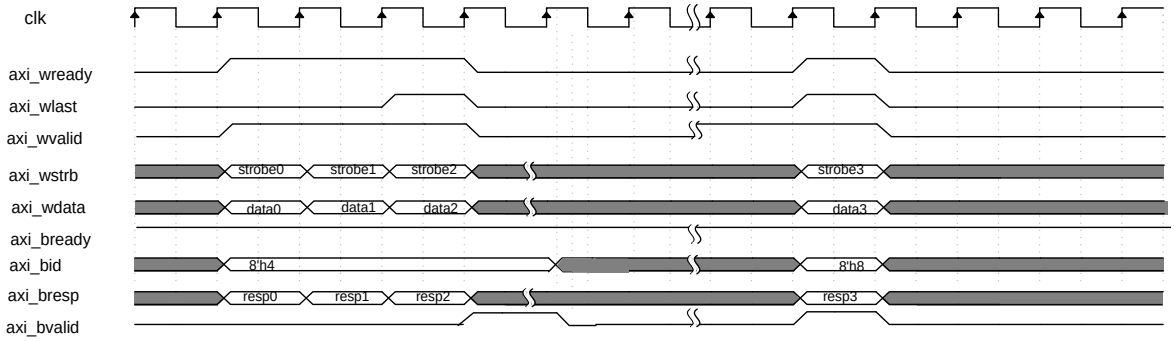

Figure 2-24 Typical Timing for Write Data

- The transmission is indicated as valid when axi_wready is valid.
- The end of a transmission is indicated when axi_wlast is valid.
- During transmission: axi_wready, axi_bid, and axi_wlast are received synchronously, while axi_wdata and axi_wstrb are sent synchronously.

2.5.2.2.4 Read data channel timing

Signals included in the read data channel: axi_rdata, axi_rready, axi_rresp, axi_rid, axi_rlast, and axi_rvalid. A typical timing is shown in Figure 2-25.

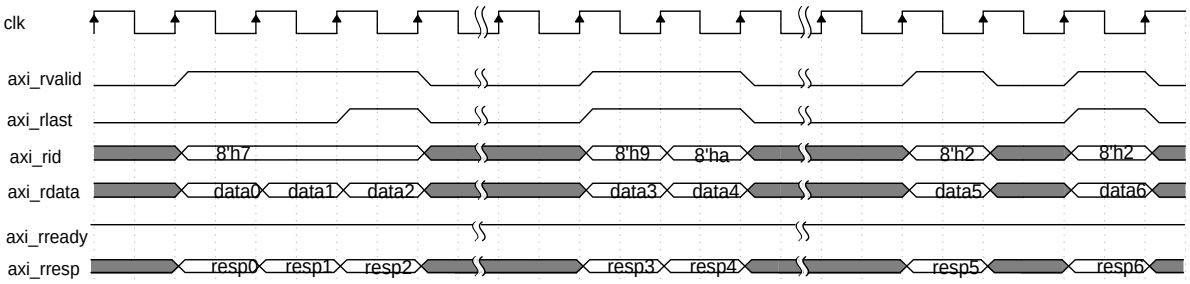

Figure 2-25 Typical Timing for Read Data

- The transmission is indicated as valid when axi_rvalid is valid.
- The end of a transmission is indicated when axi_rlast is valid.
- axi_rready needs to be pulled high whenever axi_rvalid is valid.

2.5.2.3 Config port

The Config interface uses the standard APB protocol. By reading the corresponding registers, the current state of the DDR3 can be queried.

The APB interface is for half-duplex operation communication, with independent read and write data lines and multiplexed control and address lines. Each handshake consumes at least two cycles of apb_clk.

Signals included in the APB interface: apb_enable, apb_ready, apb_rdata, apb_sel, apb_write, apb_wdata, apb_clk.

The handshake is completed when apb_enable and apb_ready are both valid simultaneously.

2.5.2.3.1 APB interface write timing

The typical APB interface write timing is as shown in Figure 2-26.

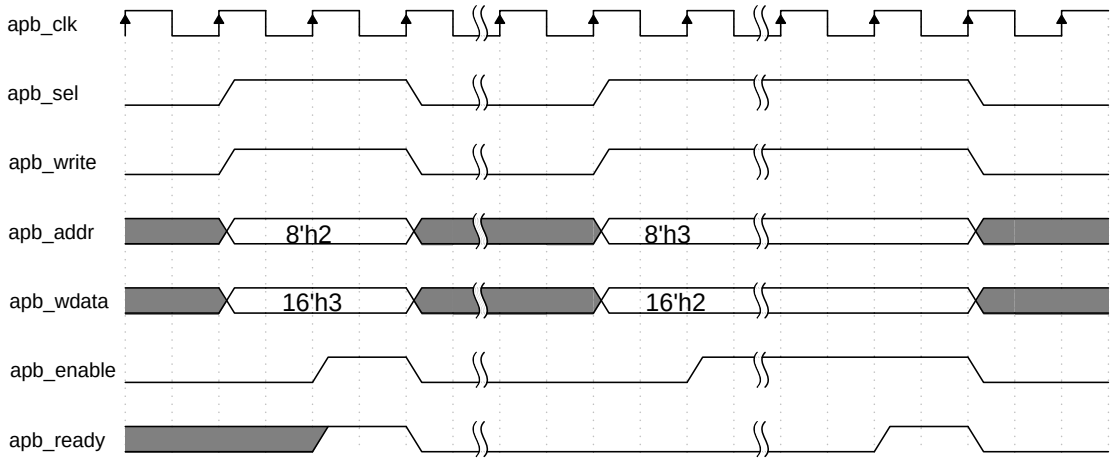

Figure 2-26 Typical Write Timing of the APB Interface

- The first clock cycle: apb_sel and apb_write are pulled high, apb_addr and apb_wdata are assigned initial values, which must remain stable until the handshake is completed and then released.
- The second clock cycle: apb_enable is pulled high until the handshake is completed and then released.

2.5.2.3.2 APB interface read timing

The typical APB interface read timing is as shown in Figure 2-27.

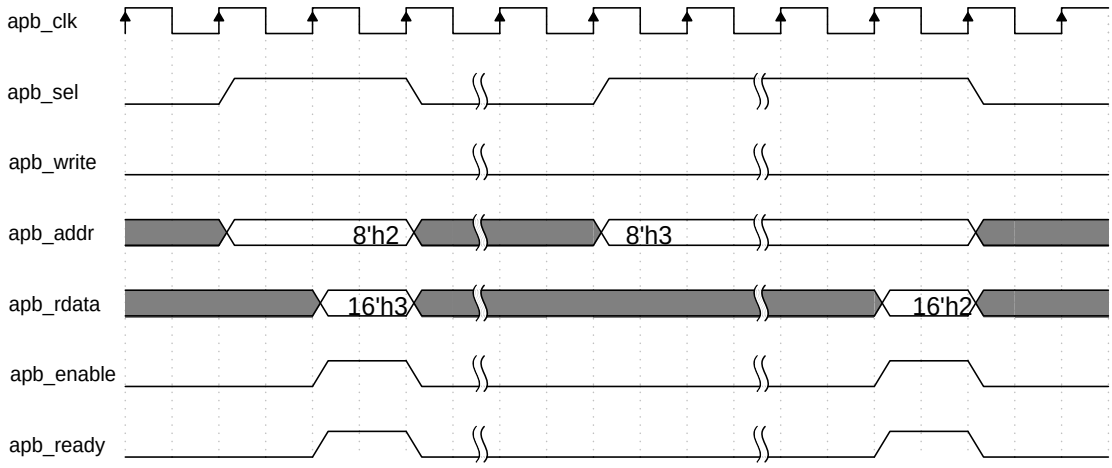

Figure 2-27 Typical Read Timing of the APB Interface

- The first clock cycle: apb_sel is pulled high, apb_write is pulled down, and apb_addr is set to an initial value, which must remain stable until the handshake is completed and then released.
- The second clock cycle: apb_enable is pulled high until the handshake is completed and then released.
- Valid data: apb_rdata is valid only during the handshake.

2.5.2.3.3 DDR3 status switching and querying method

To prevent users' status switching requests from interfering with normal DDR3 timings, certain operating rules must be followed when performing status switching (check the STATUS_REG_ADDR register to determine if the current DDR3 status and request have been responded to; control the CTRL_MODE_DATA register to achieve switching between DDR3 statuses).

- When sending a new status switch request, it is necessary to first check whether the current status switching request has been responded to, otherwise it will lead to errors;
- The status switching request and the status switching trigger enable must be sent synchronously, that is, bit 0 and bit[15:14] of CTRL_MODE_DATA must be configured at the same time.
- When configuring the Mode Registers within DDR3, the MODE_REG_0_ADDR, MODE_REG_1_ADDR, MODE_REG_2_ADDR, MODE_REG_3_ADDR registers must be configured first, followed by the CTRL_MODE_DATA register.

- After sending a status switching request, wait for two apb_clk cycles before checking the relevant status registers to refer to if a response has been received.

The state request switching process is as shown in Figure 2-28, with controlling DDR3 to enter Power Down status.

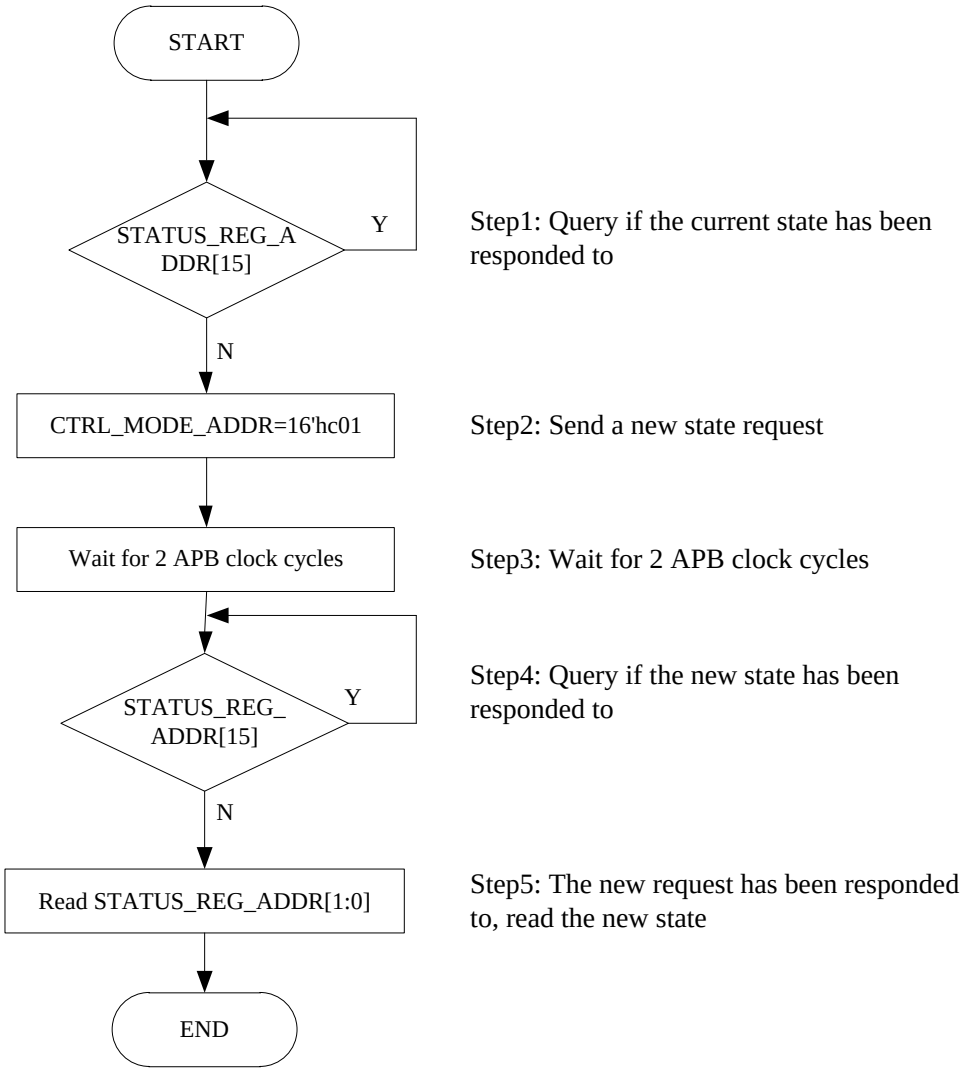

Figure 2-28 DDR3 Status Request Switching Flowchart

2.5.3 PHY Interface Description

2.5.3.1 Clock and reset interface

Table 2-20 Clock and Reset Interface

Port	I/O	Bit width	Valid Values	Description
ref_clk	I	1	-	External reference clock input.
ddr_rstn	I	1	-	External reset input.
pll_lock	O	1	High	PLL lock indication, high level indicates that it is locked.
rst_gpll_lock	O	1	High	GPLL lock indication of reset clock, high level indicates that it is locked.
ddrphy_sysclk	O	1	-	System clock of PHY, output to the working clock of the controller.

Note: "-" indicates that the parameter does not exist.

2.5.3.2 DFI interface

Table 2-21 DFI Interface

Port	I/O	Bit width	Valid Values	Description
dfi_address	I	4* MEM_ROW_WIDTH	-	DFI address bus.
dfi_bank	I	4* MEM_BANK_WIDTH	-	DFI bank address bus.
dfi_reset_n	I	4	-	DFI chip reset.
dfi_cs_n	I	4	-	DFI chip select.
dfi_ras_n	I	4	-	DFI row address strobe bus.
dfi_cas_n	I	4	-	DFI column address strobe.
dfi_we_n	I	4	-	DFI write enable signal.
dfi_cke	I	4	-	DFI clock enable.
dfi_odt	I	4	-	DFI on-die termination control bus.
dfi_rddata_valid	O	1	High	Read data valid on DFI interface.
dfi_rddata	O	DQ_WIDTH*8	-	DFI interface data.
dfi_wrdata_en	I	4	High	DFI write enable signal.
dfi_wrdata	I	DQ_WIDTH*8	-	DFI write data.
dfi_wrdata_mask	I	DQ_WIDTH*8/8	High	DFI Write data byte mask.
dfi_init_complete	O	1	High	PHY has completed training operations and is now in a normal state.
dfi_error	O	1	High	PHY training error indication.
dfi_phyupd_req	O	1	-	PHY requests an update.
dfi_phyupd_ack	I	1	-	Feedback signal of dfi_phy_req, allowing PHY to update.

Note: "-" indicates that the parameter does not exist.

2.5.3.3 Memory interface

Descirption:

The Memory interface in this IP is subject to the protocol. If the user selects a memory that includes interfaces not contained in the protocol, please refer to the corresponding module Datasheet to add and manage the interfaces properly.

Table 2-22 Memory Interface

Port	I/O	Bit width	Valid Values	Description
mem_a	O	MEM_ROW_WIDTH	-	DDR row and column address bus.
mem_ba	O	MEM_BANK_WIDTH	-	DDR Bank address.
mem_ck	O	1	-	DDR input system clock.
mem_ck_n	O	1	-	DDR input system clock.
mem_cke	O	1	High	DDR input system clock valid.
mem_dm	O	DM_WIDTH	High	DDR input data Mask.
mem_odt	O	1	-	DDR ODT.
mem_cs_n	O	1	Low	DDR chip selection.
mem_ras_n	O	1	Low	Row address enable.
mem_cas_n	O	1	Low	Column address enable.
mem_we_n	O	1	Low	DDR write enable signal.
mem_reset_n	O	1	Low	DDR reset.
mem_dq	I/O	DQ_WIDTH	-	DDR data.
mem_dqs	I/O	DQS_WIDTH	-	DDR data strobe clock.
mem_dqs_n	I/O	DQS_WIDTH	-	DDR data strobe clock.

Note: "-" indicates that the parameter does not exist.

2.5.3.4 Debug interface

Table 2-23 Debug Interface

Port	I/O	Bit width	Description
Manual Debugging Input Signal			
dbg_gate_start	I	1	Reset sequence control, active on the rising edge.
dbg_cpd_start	I	1	Reset sequence control, active on the rising edge.
dbg_ddrphy_rst_n	I	1	Reset sequence control, active low.
dbg_gpll_scan_rst	I	1	Reset sequence control, active high.

Port	I/O	Bit width	Description
force_samp_position	I	1	Enable dqsi and dqsin sampling position fixing, active high. 0: dqsi and dqsin sampling positions change during the training process. 1: dqsi and dqsin sampling positions remain unchanged, always at the initial position.
samp_position_dyn_adj	I	1	Enable dynamic adjustment after the initialization of the sampling positions of dqsi and dqsin is completed, active on the rising edge. When a rising edge is detected, an offset is added to the current sampling position, and the amount of offset is determined by init_samp_position_even and init_samp_position_odd.
init_samp_position_even	I	8*DQS_WIDTH	The offset step of dqsi initial sampling position relative to 90 °delay. The highest bit indicates the sign bit, requiring the addition operation for the value of 0 and the subtraction operation for the value of 1. There are a total of DQS_WIDTH 8-bit groups that can be configured independently. It is also valid when dynamically adjusting the dqsi and dqsin sampling positions.
init_samp_position_odd	I	8*DQS_WIDTH	The offset step of dqsi initial sampling position relative to 90 °delay. The highest bit indicates the sign bit, requiring the addition operation for the value of 0 and the subtraction operation for the value of 1. There are a total of DQS_WIDTH 8-bit groups that can be configured independently. It is also valid when dynamically adjusting the dqsi and dqsin sampling positions.
wrlvl_en	I	1	Enable write leveling, active high. 0: Skip the write leveling process, and directly use the value of init_wrlvl_step. 1:Enable write leveling.
init_wrlvl_step	I	8*DQS_WIDTH	The initial step of DQS when performing write leveling. There are a total of DQS_WIDTH 8-bit groups that can be configured independently.
wrcal_position_dyn_adj	I	1	Enable dynamic phase adjustment for DQS and DQ in the write direction, valid on the rising edge.
init_wrcal_position	I	MEM_DQS_WIDTH*8	The offset step of DQS and DQ phase initial position relative to 90 °. The highest bit indicates the sign bit, requiring the addition operation for the value of 0 and the subtraction operation for the value of 1. There are a total of DQS_WIDTH 8-bit groups that can be configured independently. It is also active when dynamically adjusting the write direction DQS and DQ phases.
force_read_clk_ctrl	I	1	Enable DQS gate position fixing, active high. 0: DQS gate position changes during the training process. 1: DQS gate position remains unchanged, always at the initial value

Port	I/O	Bit width	Description
init_slip_step	I	4*DQS_WIDTH	Initial value for dqs gate coarse adjustment position. There are a total of DQS_WIDTH 4-bit groups that can be configured independently.
init_read_clk_ctrl	I	3*DQS_WIDTH	Initial value for DQS gate fine adjustment position. There are a total of DQS_WIDTH 3-bit groups that can be configured independently.
debug_cpd_offset_adj	I	1	Enables GPLL phase adjustment, active on the rising edge. When a rising edge is detected internally, the GPLL phase is adjusted based on debug_cpd_offset_dir and debug_cpd_offset relative to the current GPLL phase.
debug_cpd_offset_dir	I	1	GPLL phase adjustment direction 0: Decrease; 1: Increase;
debug_cpd_offset	I	10	GPLL phase adjustment step.
force_ck_dly_en	I	1	Enable command and address signal output delay adjustment for the memory interface, active high. 0: The command and address signal output delay is generated by the training process. 1: The command and address signal output delay remains unchanged, always at the setting value of force_ck_dly_set_bin.
force_ck_dly_set_bin	I	8	The delay adjustment port for the command and address output signals of the memory interface. The configuration value of this port multiplied by 5ps is the additional delay value added to the command and address output signals.
Common Debugging Output Signals			
Debug Data			
debug_data	O	69*DQS_WIDTH	Debug data for each set of DDRPHY with 8-bit DQ sharing a single DDRPHY
debug_calib_ctrl	O	34	Debug data for Training status.
dbg_slice_status	O	17*MEM_DQS_WIDTH	Training status.
dbg_slice_state	O	22*MEM_DQS_WIDTH	Training status.
Others			
dbg_dll_upd_state	O	2	DLL update control state.
debug_gpll_dps_phase	O	9	GPLL current phase.
dbg_rst_dps_state	O	3	Reset clock phase adjustment status.
dbg_tran_err_rst_cnt	O	6	Reset sequence signal.
dbg_ddrphy_init_fail	O	1	DDRPHY initialization failed.
debug_dps_cnt_dir0	O	10	GPLL phase adjustment count.
debug_dps_cnt_dir1	O	10	GPLL phase adjustment count.
debug_rst_state	O	4	Reset sequence status.
debug_cpd_state	O	4	CPD alignment state.

2.5.4 PHY Interface Timing Description

2.5.4.1 DFI interface specification

When users directly access the PHY Layer, they need to comply with the DFI-like interface specification defined in this document. Through this interface, the user can perform the following operations.

- Access (read/write) DDR SDRAM;
- Put DDR3 into Power Down or Self-Refresh status;
- Dynamically configure the values of the internal registers in DDR SDRAM;
- Read the status of the DDR SDRAM.

The DFI interface used in this design differs from the standard "***DFI 3.1 Specification***", and differences can be seen in Table 2-24 and Table 2-25.

Table 2-24 Differences between the DFI Interface of this Design and Standard DFI

Interface	**Differences**
ControlInterface	No requirements for tctrl_delay and tcmd_lat.
Write DataInterface	Except for the dfi_wrdata_cs_n signal, it is compatible with the standard DFI.
Read DataInterface	Only the dfi_rddata and dfi_rddata_valid signals are retained.
UpdateInterface	A custom interface is used.
StatusInterface	Only dfi_init_complete is retained.
TrainingInterface	N/A.
Low PowerControl Interface	N/A.
ErrorInterface	Compatible.

Table 2-25 Detailed Comparison of Differences between DFI Interface and Standard DFI

Interface Signal Classification	Standard DFI	Simplified DFI	Signal Source	Description
Control signals	dfi_address	dfi_address	MC	DFI address bus.
	dfi_bank	dfi_bank	MC	DFI bank address bus.
	dfi_cas_n	dfi_cas_n	MC	DFI column address strobe, only applicable to DDR3.
	dfi_cid	N/A	MC	DFI Chip ID.
	dfi_cke	dfi_cke	MC	DFI clock enable.
	dfi_cs_n	dfi_cs_n	MC	DFI chip select.
	dfi_odt	dfi_odt	MC	DFI on-die termination control bus
	dif_ras_n	dif_ras_n	MC	DFI row address strobe bus, only applicable to DDR3.
	dfi_reset_n	dfi_reset_n	MC	DFI chip reset.
	dfi_we_n	dfi_we_n	MC	DFI write enable signal, only applicable to DDR3.
Write Data Siganls	dfi_wrdata	dfi_wrdata	MC	DFI write data.
	dfi_wrdata_cs_n	N/A	MC	DFI Write Data Chip Select.
	dfi_wrdata_en	dfi_wrdata_en	MC	DFI write enable signal.
	dfi_wrdata_mask	dfi_wrdata_mask	MC	DFI Write data byte mask.
Read Data Signals	dfi_rddata	dfi_rddata	PHY	DFI interface data.
	dfi_rddata_cs_n	N/A	MC	DFI Read Data Chip Select.
	dfi_rdata_dbi_n	N/A	PHY	Read Data DBI.
	dfi_rddata_en	N/A	MC	Read data enable.
	dfi_rddata_valid	dfi_rddata_valid	PHY	Read data valid on DFI interface.
	dfi_rddata_dnv	N/A	PHY	DFI data not valid.
Update Signals	dfi_ctrlupd_ack	N/A	PHY	MC-initiated update acknowledge.
	dfi_ctrlupd_req	N/A	MC	MC-initiated update request.
	dfi_phyupd_ack	dfi_phyupd_ack	MC	PHY-initiated update acknowledge.
	dfi_phyupd_req	dfi_phyupd_req	PHY	PHY-initiated update request.
	dfi_phyupd_type	N/A	PHY	PHY-initiated update select.
Status Signals	dfi_alert_n	N/A	PHY	CRC or parity error indicator.
	dfi_data_byte_disable	N/A	MC	Data byte disable.
	dfi_dram_clk_disable	N/A	MC	DRAM clock disable.
	dfi_freq_ratio	N/A	MC	DFI frequency ratio indicator.
	dfi_init_complete	dfi_init_complete	PHY	PHY initialization complete.
	dfi_init_start	N/A	MC	DFI setup stabilization or frequency change initiation.
	dfi_parity_in	N/A	MC	Parity value.

Interface Signal Classification	Standard DFI	Simplified DFI	Signal Source	Description
DFI Training Signals	dfi_calvl_capture	N/A	MC	CA training capture.
	dfi_phy_calvl_cs_n	N/A	PHY	CA training chip select.
	dfi_calvl_en	N/A	MC	PHY CA training logic enable.
	dfi_calvl_req	N/A	PHY	PHY-initiated CA training request.
	dfi_calvl_resp	N/A	PHY	CA training response.
	dfi_lvl_pattern	N/A	MC	Training pattern.
	dfi_lvl_periodic	N/A	MC	Training length indicator (full or periodic).
	dfi_phylvl_ack_cs_n	N/A	MC	DFI PHY training chip select acknowledge.
	dfi_phylvl_req_cs_n	N/A	PHY	DFI PHY training chip select request.
	dfi_phy_rdlvl_cs_n	N/A	PHY	Read training chip select for read data eye training.
	dfi_phy_rdlvl_gate_cs_n	N/A	PHY	Read training chip select for gate training.
	dfi_phy_wrlvl_cs_n	N/A	PHY	Write leveling chip select.
	dfi_rdlvl_en	N/A	MC	PHY read data eye training logic enable.
	dfi_rdlvl_gate_en	N/A	MC	PHY gate training logic enable.
	dfi_rdlvl_gate_req	N/A	PHY	PHY-initiated gate training request.
	dfi_rdlvl_req	N/A	PHY	PHY-initiated read data eye training request.
	dfi_rdlvl_resp	N/A	PHY	Read training response.
	dfi_wrlvl_en	N/A	MC	PHY write leveling logic enable.
	dfi_wrlvl_req	N/A	PHY	PHY write leveling request.
	dfi_wrlvl_resp	N/A	PHY	Write leveling response.
	dfi_wrlvl_strobe	N/A	MC	Write leveling strobe.
Low Power Control Signals	dfi_lp_ack	N/A	PHY	Low power acknowledge
	dfi_lp_ctrl_req	N/A	MC	Low power opportunity control request.
	dfi_lp_data_req	N/A	MC	Low power opportunity data request.
	dfi_lp_wakeup	N/A	MC	Low power wakeup time.
Error Signals	dif_error	dif_error	PHY	DFI Error.
	dfi_error_info	N/A	PHY	DFI Error Info.

2.5.4.2 DFI interface timing

The working clock frequency ratio of MC to PHY is 1:4, meaning each MC instruction corresponds to 4 Phase PHY instructions. Instructions and data can be arranged across the 4 Phases of PHY as

required. When sending control instructions through the DFI interface, the user must strictly adhere to the DDR SDRAM requirements for instruction and data timing delays. DFI instructions can only be received after dfi_init_complete is pulled high, with PHY data alignment for read operations.

2.5.4.2.1 DFI interface write timing

Control lines for write operations: dfi_cs_n, dfi_ras_n, dfi_cas_n, dfi_we_n, dfi_cke, dfi_odt.

Address lines for write operations: dfi_bank, dfi_address.

Data lines for write operations: dfi_wrdata_en, dfi_wrdata, dfi_wrdata_mask.

To write data to DDR3 SDRAM, a write instruction must be sent first, followed by data, with typical timing as shown in Figure 2-29.

2.5.4.2.2 DFI interface read timing

Control lines for read operations: dfi_cs_n, dfi_ras_n, dfi_cas_n, dfi_we_n, dfi_cke, dfi_odt.

Address lines for read operations: dfi_bank, dfi_address.

Data lines for read operations: dfi_rddata, dfi_rddata_valid.

To read data from DDR3 SDRAM, a read instruction must first be sent, and then data is received from the PHY via DFI, with typical timing as shown in Figure 2-30.

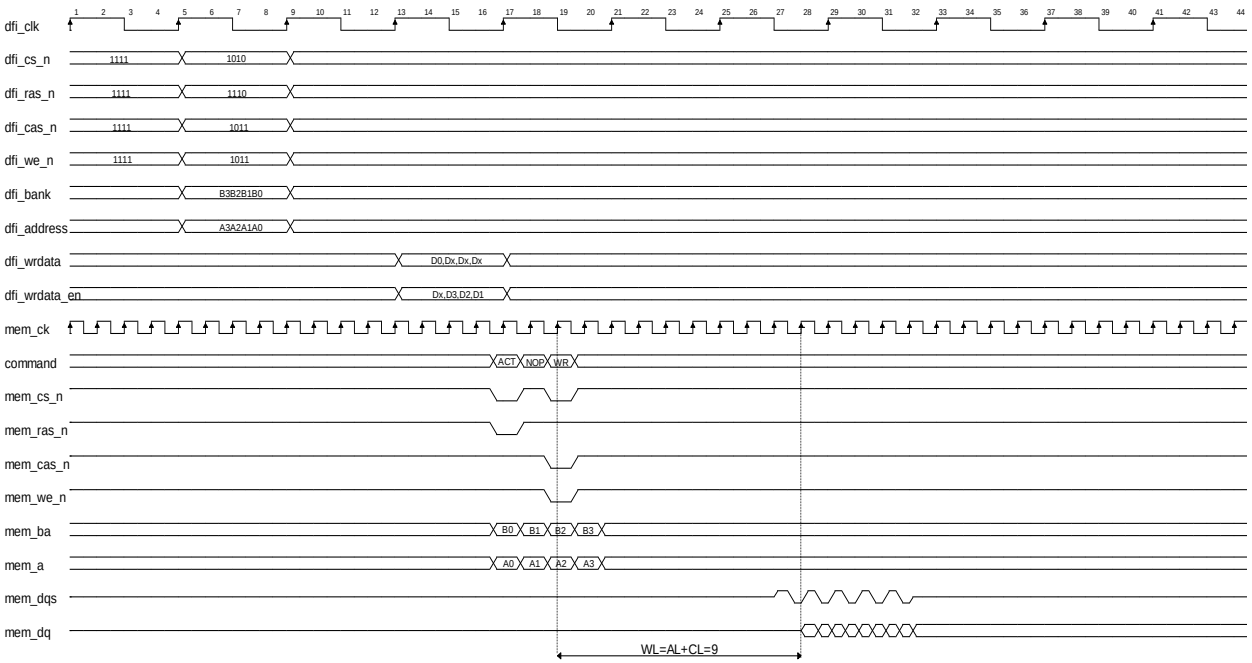

Figure 2-29 DFI Interface Write Operation Timing

Note: This timing diagram is a simulation with WL=9.

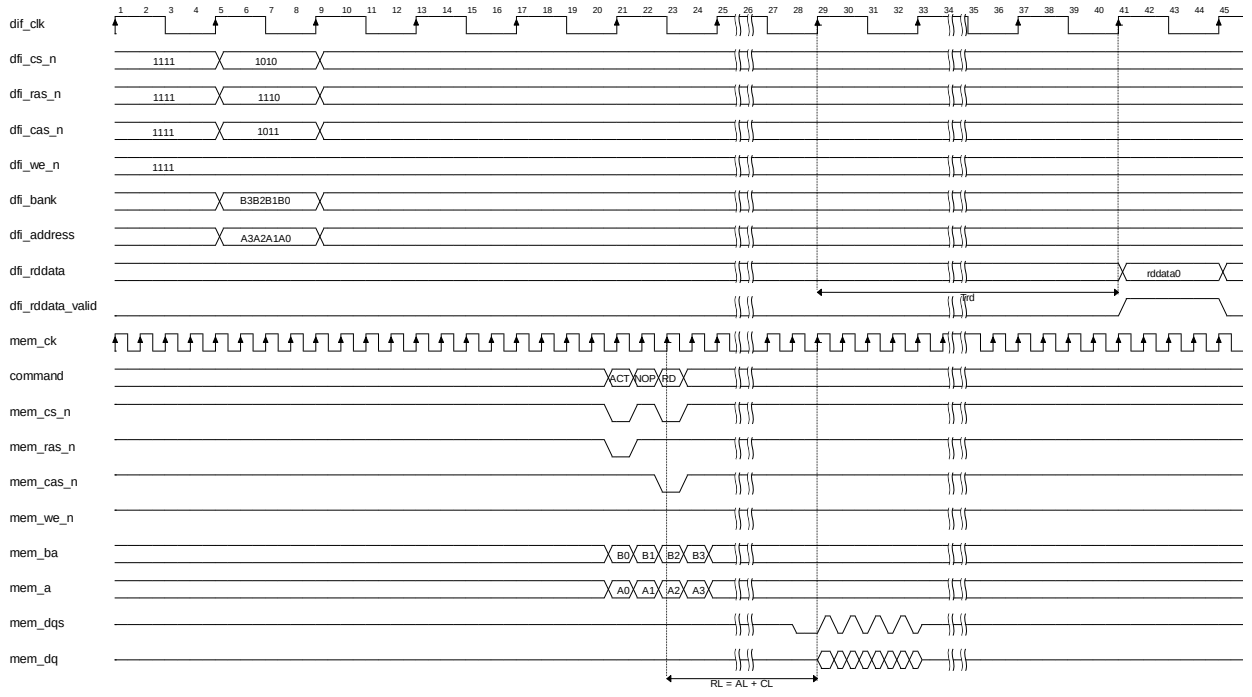

Figure 2-30 DFI Interface Read Operation Timing

Note: This timing diagram is a simulation with Trd of 7 dfi_clk cycles; the actual on-board timing may differ from the above.

2.5.5 Debug Data Description

2.5.5.1 debug_data description

For the description of the meanings of each field of debug_data, please refer to Table 2-26.

Table 2-26 Definitions of debug_data Fields

Bits	Debug Port	Description
3:0	coarse_slip_step	Coarse adjustment position obtained from dqs gate training
6:4	read_clk_ctrl	Fine adjustment position obtained from dqs gate training
10:7	gate_win_size	Number of valid windows for dqs gate training
11	gate_check_pass	DQS gate training completion indicator (set to 0 after completion)
12	rddata_check_pass	Read calibration data completion confirmation indicator (set to 0 after completion)
20:13	dqs_even_bin	DQS rising edge delay step value obtained after read eye calibration completion
28:21	dqs_odd_bin	DQS falling edge delay step value obtained after read eye calibration completion
36:29	total_margin_even	Margin for DQS rising edge window
45:38	total_margin_odd	Margin for DQS falling edge window
55:48	wrlvl_step	dqs delay steps obtained when write leveling is complete
56	Reserved	Reserved
57	wrlvl_dq	DQ value returned during write leveling
65:58	wl_p_dll_bin	Write leveling delay code plus 90-degree code
66	this_group_ca_dly	The number of mem_ck clock cycles that the CA signals of this group are delayed.
68:67	ck_dqs_diff	The difference between the mem_ck clock cycles for the CA and DQS signals of this group

2.5.5.2 Description of debug_calib_ctrl

For the description of debug_calib_ctrl fields, please refer to Table 2-27.

Table 2-27 Definitions of debug_calib_ctrl Fields

Bits	Debug Port	Description
0	calib_error	Training error indicator, generated due to read calibration errors during write leveling
4:1	dbg_upcal	Update control state machine
8:5	dbg_eyecal	Read eye calibration control state machine
12:9	dbg_wrcal	Write calibration control state machine
17:13	dbg_rdcal	MPR read calibration control state machine
21:18	dbg_wrlvl	write leveling calibration control state machine
25:22	dbg_init	Power-up initialization process control state machine
29:26	dbg_main	Master state machine for training

Bits	Debug Port	Description
33:30	dbg_error_status	The training state when an error occurs. 0: No error; 1: Write leveling generates an error; 2: Read pattern generates an error; 3: Gate training generates an error; 4: Read training generates an error; 5: Write training generates an error; 6: Eye training generates an error.

2.5.5.3 dbg_slice_status Description

The bit width of dbg_slice_status is 4*MEM_DQS_WIDTH+12. For MEM_DQS_WIDTH of 1, please refer to the description of each field in Table 2-28.

Table 2-28 Definitions of dbg_slice_state Fields

Bits	Debug Port	Description
0	dll_update_code_done	Indicator signal for the completion of DLL update. 1: Completed; 0: Not completed.
1	eyecal_move_done	Indicator signal for the completion of a single delay adjustment in eye diagram calibration. 1: Adjustment completed; 0: Adjustment not completed;
2	eyecal_check_pass	Indicator signal for the completion of eye diagram calibration 1: Eye diagram calibration completed; 0: Eye diagram calibration not completed.
3	wrcal_move_done	Indicator signal for the completion of a single delay adjustment in write calibration. 1: Adjustment completed; 0: Adjustment not completed;
4	wrcal_check_pass	Indicator signal for the completion of write calibration. 1: Write calibration completed; 0: Write calibration not completed;
5	dll_lock_tmp	DLL lock indicator, with each bit corresponding to a group, and higher bits for higher groups. 1: Locked; 0: Unlocked
6	dqs_gate_comp_done	Indicator signal for the completion of DQS gate update. 1: Completed; 0: Not completed.
7	rddata_check_pass_tmp	Data comparison passed after DQS gating, with each bit corresponding to a group; higher bits correspond to higher groups. 1: Passed; 0: Not passed;
8	gate_cal_error	Indicator signal for DQS gate error. 1: DQS gate error; 0: No DQS gate error;
9	gate_adj_done	Indicator signal for the completion of a single position adjustment for DQS gate. 1: Adjustment completed; 0: Adjustment not completed;
10	gate_check_pass	Indicator signal for the completion of DQS gate. 1: DQS gate completed; 0: DQS gate not completed;
11	rdel_move_done	Indicator signal for the completion of a single delay adjustment in read calibration. 1: Adjustment completed; 0: Adjustment not completed;
12	rdel_calib_error	Indicator signal for read calibration error. 1: Calibration error; 0: No error;

Bits	Debug Port	Description
13	rdel_calib_done	Indicator signal for the completion of read calibration. 1: Calibration completed; 0: Calibration not completed.
14	adj_rdel_done	Indicator signal for the completion of initial adjustment in read calibration. 1: Adjustment completed; 0: Adjustment not completed;
15	wrlvl_dqs_resp_tmp	Write leveling calibration completion indicator, with each bit corresponding to a group, and higher bits for higher groups. 1: Calibration completed; 0: Calibration not completed.
16	wrlvl_error_tmp	Write leveling calibration error indicator, with each bit corresponding to a group, and higher bits for higher groups. 1: Calibration error; 0: No error;

2.5.5.4 Description of dbg_slice_state

For the description of dbg_slice_state fields, please refer to Table 2-29.

Table 2-29 Definitions of dbg_slice_state Fields

Bits	Debug Port	Description
7:0	dbg_slice_rdchk_state	State output of the read calibration data comparison state machine.
11:8	dbg_slice_rpbd_ctrl_state	State output of the read calibration control DQS delay state machine.
14:12	dbg_slice_gate_state	State output of the adjustment of DQS gate delay state machine.
18:15	dbg_slice_dq_wrcal_state[3:0]	State output of the control write direction DQ delay state machine.
21:19	dbg_slice_wrlvl_state	State output of the control state machine for write leveling delay.

2.6 Description of the IP Register

This section provides the HMIC_S IP related register description and access methods.

2.6.1 Register Description

2.6.1.1 Internal Register of the Controller

2.6.1.1.1 MODE_REG_0_ADDR

The definition of each bit in this register is the same as MR0 in "***DDR3 SDRAM Standard***", with a bit width of 16 bits and an access address of 0x00.

Table 2-30 Definitions of MODE_REG_0_ADDR Bits

Bits	Item	Reset Values	Access Type	Description
1:0	BL	2'b00	R	Burst Length.
2	CL[0]	Instantiation value	R/W	CAS Latency, CL bit 0.
3	RBT	Instantiation value	R/W	Read Burst Type.
6:4	CL[3:1]	Instantiation value	R/W	CAS Latency, CL bit [3:1], with a maximum value of 8.
7	TM	1'b0	R	Mode.
8	DLL	0	R	DLL Reset.
11:9	WR	Instantiation value	R/W	Write recovery for autoprecharge.
12	PPD	1	R	DLL Control for Precharge PD.
15:13	Reserved	0	R	Reserved.

2.6.1.1.2 MODE_REG_1_ADDR

The definition of each bit in this register is the same as MR1 in "***DDR3 SDRAM Standard***", with a bit width of 16 bits and an access address of 0x01.

Table 2-31 Definitions of MODE_REG_1_ADDR Bits

Bits	Item	Reset Values	Access Type	Description
0	DLL	1'b0	R	DLL Enable.
1	D.I.C[0]	1'b0	R/W	Output Driver Impedance Control, D.I.C bit 0.
2	Rtt_Nom[0]	Instantiation value	R/W	Rtt_Nom, bit 0.
4:3	AL	2'b10	R	Additive Latency.
5	D.I.C[1]	Instantiation value	R/W	Output Driver Impedance Control, D.I.C bit 1.
6	Rtt_Nom[1]	1'b0	R/W	Rtt_Nom bit 1.
7	Level	1'b0	R	Write leveling enable. For detailed values, please refer to JESD79-3E.
8	Reserved	1'b0	R	Reserved.
9	Rtt_Nom[2]	Instantiation value	R/W	Rtt_Nom bit 2.
10	Reserved	1'b0	R	Reserved.
11	TDQS	1'b0	R	TDQS enable.
12	Qoff	1'b0	R	Qoff.
15:13	Reserved	3'b000	R	Reserved.

2.6.1.1.3 MODE_REG_2_ADDR

The definition of each bit in this register is the same as MR2 in "***DDR3 SDRAM Standard***", with a bit width of 16 bits and an access address of 0x02.

Table 2-32 Definitions of MODE_REG_2_ADDR Bits

Bits	Item	Reset Values	Access Type	Description
2:0	PASR	3'b0	R	Partial Array Self-Refresh (Optional).
5:3	CWL	Instantiation value	R/W	CAS write Latency.
6	ASR	1'b0	R	Auto Self-Refresh.
7	SRT	1'b0	R	Self-Refresh Temperature Range.
8	Reserved	1'b0	R	Reserved.
10:9	Rtt_WR	2'b00	R/W	Rtt_WR.
15:11	Reserved	5'b0	R	Reserved.

2.6.1.1.4 MODE_REG_3_ADDR

This register stores all configuration bits of MR3, with a bit width of 16 bits and an access address of 0x03. Refer to Table 2-33 for the definition of each Bit.

Table 2-33 Definitions of MODE_REG_3_ADDR Bits

Bits	Item	Reset Values	Access Type	Description
[15:0]	Reserved	16'h0	R	For detailed values, please refer to JESD79-3E.

2.6.1.1.5 CTRL_MODE_DATA

This register stores user-issued instructions (MRS, Low Power, Normal) with an access address of 0x04. Refer to Table 2-34 for the definition of each Bit.

Table 2-34 Definitions of CTRL_MODE_DATA Bits

Bits	Item	Reset Values	Access Type	Description
0	enable	1'b0	R/W	Enable DDR core status switching trigger.
13:1	Reserved	13'h0	R	Reserved.
15:14	select	2'b00	R/W	DDR Core status selection. 00: Normal; 01: MRS; 10: Self-Refresh; 11: Power Down.

2.6.1.1.6 STATUS_REG_DATA

This register stores the current status of DDR, with an access address of 0x05. Refer to Table 2-35 for the definition of each Bit.

Table 2-35 Definitions of STATUS_REG_DATA Bits

Bits	Item	Reset Values	Access Type	Description
1:0	mode_state	2'b0	R	Stores the current status of DDR3. 2'b00: Normal State; 2'b01:Self-Refresh State; 2'b10:Power Down State; 2'b11:MRS State.
14:2	Reserved	13'b0	R	Reserved
15	busy	1'b0	R	Whether the current request of DDR3 has been responded to: 1'b0: Current request has been responded to 1'b1: Current request has not been processed

2.6.1.2 PHY Internal Registers

DDR PHY internal registers support DDR3, DDR2, and LPDDR, and are fully consistent with the protocol. For detailed descriptions, please refer to "***JESD79-3D, DDR3 SDRAM Standard***"; "***JESD79-2F, DDR2 SDRAM SPECIFICATION***"; "***JESD209B, "Low Power Double Data Rate (LPDDR) SDRAM Standard***".

2.6.2 Register Access

The internal registers of the Controller are accessed via the APB interface, with register data transferred through apb_wdata and apb_rdata, and the register address through apb_addr. Refer to "2.6.1 Register Description" for the address values of each register.

The method for accessing the PHY internal registers is consistent with the method for accessing the Memory internal Mode Register.

2.7 Typical Applications

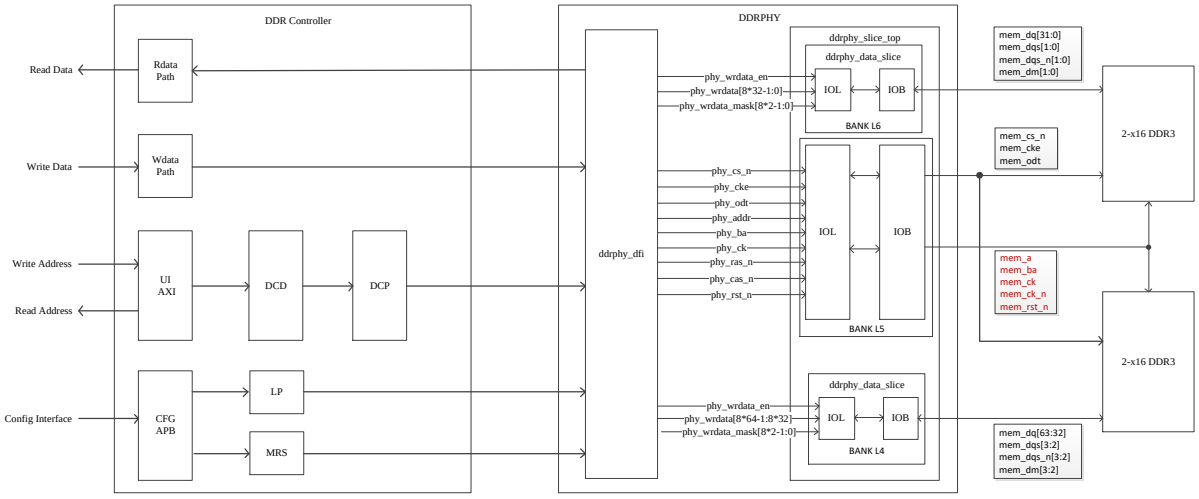

Figure 2-31 Multi-bank-x64 Structure Diagram

The typical applications of Multi-Bank-x64 are shown in Figure 2-31. The Multi-bank-x64 is based on two adjacent HR banks (shown as L4, L5, L6), where Bank L5 is used for the CA channel, and Bank L4 and Bank L6 are used for the DQ channel. The following precautions apply to Multi-Bank-x64 applications:

- Do not use the two dedicated pins for calibration resistors within the bank.
- It is recommended to distribute the CA signal across Bank L5 to ensure a minimal delay difference in the clock distribution path.
- In the Fly-by topology, ensure that the far-end stub tracing conforms to the design requirements as much as possible to cover AC parameters.

2.8 Descriptions and Considerations

2.8.1 AXI4 Interface Burst Calculation

Start_Address = ADDR;

Burst_Length= len;

End_Address = ADDR + len * 8.

2.8.2 Column Address Rounding Examples

For example, if the column address width of DDR3 SDRAM is 10, the user's access request length is 2, and the starting address of the column address is 10'h3F8, then the address for the second access will be 10'h000.

2.8.3 Clock Constraints

Multiple clocks within the HMIC_S IP require constraints, namely ref_clk, rst_clk, ddrphy_sysclk, and phy_dq_sysclk (one constraint required for each bank). The relation between these clocks is shown in Figure 2-32. For specific constraint methods, please refer to the .fdc files in the "IP /pnr" directory.

Among them, the ref_clk is the input reference clock. ddrphy_sysclk and the rst_clk are obtained by GPLL frequency multiplication, and phy_dq_sysclk is obtained by PPLL frequency multiplication. ddrphy_sysclk is the system clock for the IP soft logic, rst_clk is the drive clock for the GPLL dynamic phase shift and sequence reset, and phy_dq_sysclk is the bank clock. Each bank has a PPLL, and phy_dq_sysclk is obtained by the PPLL division in that bank.

Attention:

The timing constraints for ddrphy_sysclk and phy_dq_sysclk provided in the .fdc file are by default supported only by the ads synthesis tool. To use the OEM synthesis tool, the constraints must be switched to the OEM format constraints stated in the .fdc file.

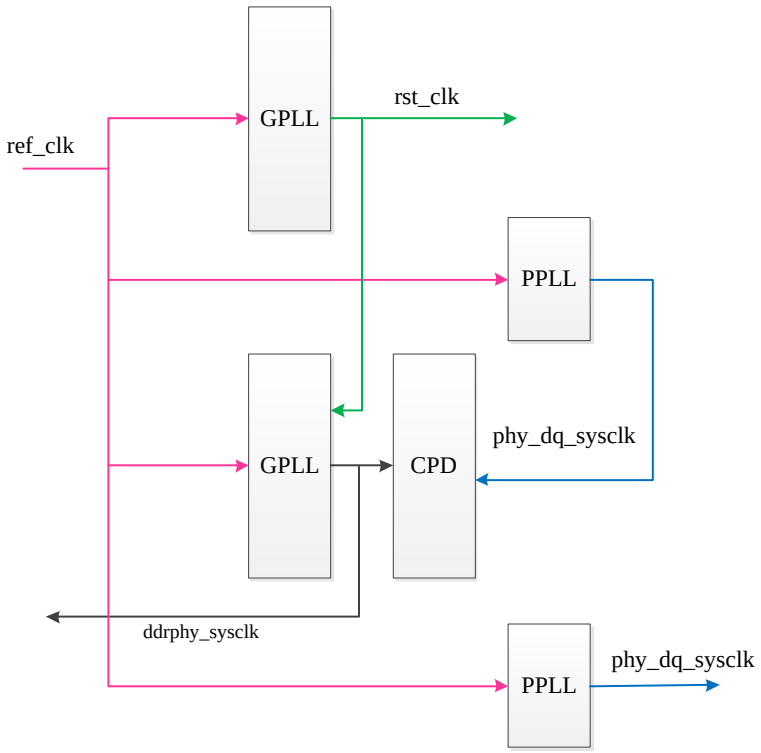

Figure 2-32 IP Internal Clock Structure

2.8.4 IO Constraints

Position constraints for the chip's internal modules must be consistent with the location constraints provided by the .fdc file, in which the constraint location cannot be changed, and the routing should be modified according to specific usage.

Constraints on pin positions and level standards, among other parameters, are set in the UCE (User Constraint Editor), for which specific configuration can be referred to in the .fdc files in the "IP /pnr" directory. The requirements for pin parameter settings are as follows.

- VCCIO needs to be set to 1.5V or 1.8V (1.5V for DDR3, and 1.8V for DDR2 and LPDDR);

- mem_dqs is a bidirectional differential signal in DDR3 and DDR2 modes and requires SSTL15D_I or SSTL18D_I levels; in LPDDR mode, it is a single-end signal and requires SSTL18_I level;
- Other signals are single-end signals and require SSTL15_I and SSTL18 levels.
- mem_dq is a bidirectional single-end signal and requires the configuration of reference voltage VREF. By default, the internal VREF is used, with the VREF_MODE set to INT;
- mem_dqs/mem_dq signals need termination resistors at the FPGA end, and it is recommended to use internal termination resistors by setting DDR_TERM_MODE to ON and CP_DYN_TERM also to ON;
- All other parameters can use the default settings.

2.8.5 Route Constraints

If the DDR project requires priority route constraints, the priority route information is stored in the inst_name.rcf file in the "/pnr" directory; the Example Design project will automatically load this file. If users create their own project, they need to manually load the file. You can refer to the help document under the PDS installation path: "***Route_Constraint_Editor_User_Guide***".

2.8.6 SCBV Setting

When using PG2L100H and PG2T390H products, if any L4 Bank or L5 Bank is unused, SCBV needs to be set. For the setting rules, please refer to Table 2-36.

Table 2-36 SCBV Setting Rules

Applicable Devices	**L4 Bank**	**L5 Bank**	**SCBV Setting**
PG2L100H PG2T390H	Used	Not used	High Voltage (V≥2.5V) Low Voltage (V<2.5V)
	Not used	Used	
	Not used	Not used	High Voltage/Low Voltage

Note: V is the operating voltage of the bank in use.

The SCBV setting interface is shown in Figure 2-33. In Project Setting, select Configuration under the Generate Bitstream tab to see the Set Configuration Bank Voltage (SCBV) option, which is divided into Low Voltage and High Voltage. For detailed information, please refer to

"***UG050012_Titan2 Single Board Hardware Design Guide***" and "***UG040012 Logos2 Single Board Hardware Design User Guide***".

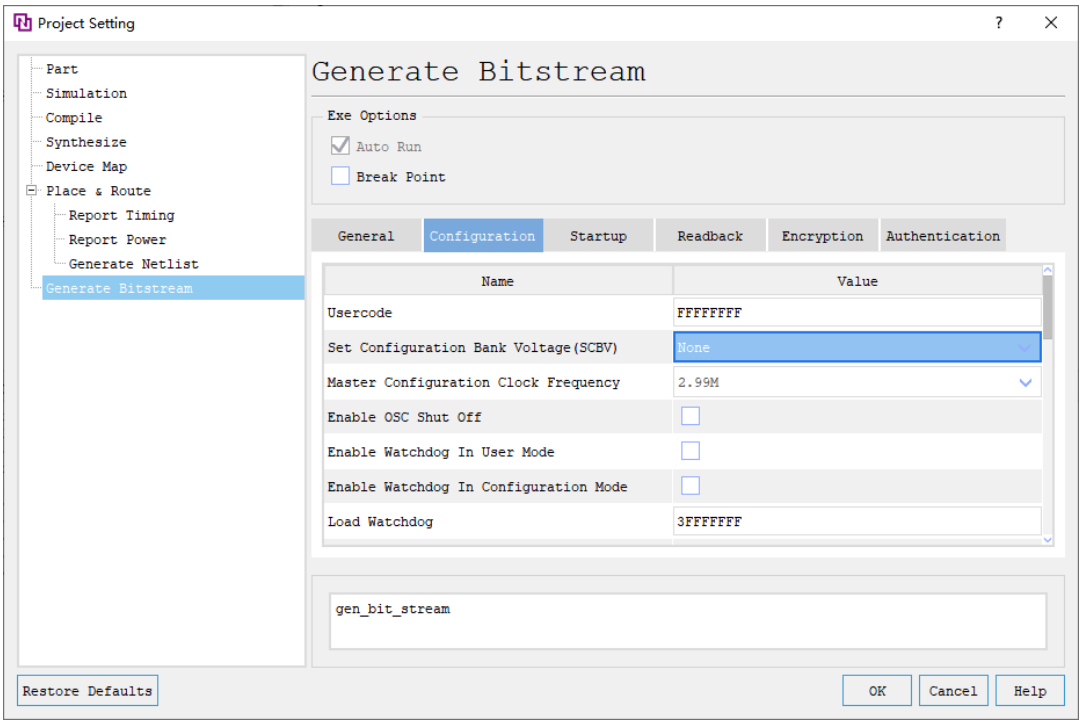

Figure 2-33 SCBV Setting Interface

2.9 IP Debugging Method

2.9.1 Key Indicator Signal

For key information during the operation of DDRPHY, single-bit indicators are created for easy observation, which can be connected to external LEDs or monitored in other ways to quickly determine the operating status of DDRPHY. For descriptions of key indicator signal, please refer to Table 2-37.

Table 2-37 Key Indicator Signal

Port	I/O	Bit width	Description
ddrphy_cpd_lock	O	1	CPD lock indication, a high level indicates that it is locked.
ddr_init_done	O	1	PHY has completed training operations and is now in a normal state.
heart_beat_led	O	1	Heartbeat signal. When the ddrphy system clock is normal, it flashes approximately once per second. (Signal generated in Example Design)
err_flag_led	O	1	Data detection error signal Flashing: BIST is operating normally, data monitoring is error-free. 1: Data errors detected, which need to be manually cleared after troubleshooting. 0: BIST fails to operate normally. (Signal generated in Example Design)

2.9.2 Internal status and control signals

For other internal status signals and control signals used for debugging that do not change in real-time, please refer to Table 2-23. These signals can be conveniently read and written through the Serial Port. Flexible debugging can be achieved by developing accompanying scripts.

For a serial port access diagram, please refer to Figure 2-14. In the DDRPHY Example Design, the ips2l_uart_ctrl module is a serial port communication module, and the host computer reads and writes internal status signals and control signals through the serial port and ips2l_uart_ctrl module.

For the Serial Port configuration used in DDRPHY Example Design, please refer to Table 2-38.

Table 2-38 DDRPHY Example Design Serial Port Configuration

Baud Rate	Start Bit	Data Bits	Stop Bits	Checksum Bits	Flow Control
115200bps	1bit	8bit	1bit	None	None

Descirption:

A set of Serial Port scripts was developed during the internal debugging process. Please contact AE to obtain if needed.

Disclaimer

Copyright Notice

Disclaimer

1. This document only provides information in stages and may be updated at any time based on the actual situation of the products without further notice. The Company assumes no legal responsibility for any direct or indirect losses caused by improper use of this document.
2. This document is provided "as is" without any warranties, including but not limited to warranties of merchantability, fitness for a particular purpose, non-infringement, or any other warranties mentioned in proposals, specifications, or samples. This document does not grant any explicit or implied intellectual property usage license, whether by estoppel or otherwise.
3. The Company reserves the right to modify any documents related to its family's products at any time without prior notice.