HMIC_S IP User Guide

(UG042003,V1.13b)

(2024-03-28)

Shenzhen Ziguang Tongchuang Electronics Co., Ltd.

Document version revision history

Date	Version	Revision history	Applicable IP and corresponding versions
2020-04-27	V1.0	1. Initial version.	V1.0, V1.1
2020-06-04	V1.1	1. Update interface screenshot; 2. Corrected the maximum speed of DDR3 to 800Mbps; 3. Add description of position constraints in Notes.	V1.2
2020-07-16	V1.2	1. Update interface screenshot; 2. Corrected DDR3 maximum speed to 1066Mbps.	V1.2
2020-09-25	V1.3	1. Update interface screenshot; 2. Update debug interface signal description; 3. Add VCS simulation instructions; 4. Updated reference design block diagram.	V1.3
2020-10-13	V1.4	1. Add support for multiple packages.	V1.4
2020-11-12	V1.5	1. Update interface screenshot; 2. Controller interface update; 3. Add PLL reference clock bank selection; 4. Delete the dbg_dll_update_en port; 5. Add precautions.	V1.5, V1.5a
2020-12-12	V1.5a	1. In Read/Write address channel, the description related to the packet length is updated.	V1.5a
2020-12-29	V1.6	1. Update interface screenshot; 2. Add support for two packages of 330H.	V1.6
2021-07-20	V1.6a	1. Update interface screenshot; 2. Modify the device name, 330H is changed to 390H.	V1.6a
2021-12-03	V1.6b	1. Adjust the manual structure and standardize the format; 2. Add "Applicable IP and corresponding versions" to the document version revision record. "this" description column; 3. Add IP version change record; 4. Improve the description of module instantiation and update the screenshot of the IP selection path interface, see "2.3.1 Module Instantiation"; 5. Update the screenshot of the Pin/Bank Options page, add the "Enable fdc file select" configuration parameter, and delete the default CA_GROUP selected interface option, see Figure 2-8 and Table 2-4; 6. Improve the description of the default value of DDR3 related interface configuration parameters, add the description of the related default values of DDR2 and LPDDR, update the list of configuration parameter descriptions, and add the "IP configuration interface default value" column, see Table 2-2, Table 2-3, Table 2-4; 7. Update the screenshot of the Summary page and add the description of reference voltage and particle capacity, see Figure 2-9; 8. Improve the output file description after IP	V1.6b

		generation, see Table 2-5; 9. Add "2.3.5 Resource Utilization"; 10. For detailed comparison between newly added Simplified AXI4 and standard AXI4, see "2.5.2 1 Simplified AXI4 Interface"; 11. Correct the packet length description related to Simplified AXI4 interface address channel read/write timing, see "2.5.2 1(1) Write address channel timing" and "2.5.2 1(2) Read address channel timing"; 12. Improve the DFI interface list and Memory interface list, update the bit width parameter names "MEM_ROW_WIDTH" and "MEM_BANK_WIDTH", see Table 2-17 and Table 2-18; 13. Update the debug interface list, add the ck delay top-level configurable port "ck_dly_set_bin", and improve the Debug interface signal classification, see Table 2-19; 14. For a detailed comparison of the differences between the new DFI interface and the standard DFI, see "2.5.4 1 DFI Interface Specification"; 15. Improve Debug data description, see "2.5.5 Debug Data Description"; 16. Add "2.6.2 Register Access"; 17. Add typical applications in multi-BANK-x64 scenarios, see "2.7 Typical Applications"; 18. Improve the description of IO constraints related to mem_dqs and mem_dq, see "2.8.4 IO Constraints"; 19. Add "2.8.5 IP debugging methods"; 20. Added new reference documents, glossary, abbreviation list, and statement-related content, see "Chapter 3 Appendix".	
2022-04-29	V1.7	1. Add the IP version change record corresponding to the V1.7 version; 2. Add support for PG2L25H and PG2L50H devices, and update the applicable device list and resource utilization list simultaneously, see Table 2-1 and Table 2-6; 3. In order to optimize the GUI interface, update the SDRAM particle model, add the "Enable CS_n" option, add the description related to the CA signal legality check, and simultaneously update the Memory Options, Pin/Bank Options and Summary page screenshots and corresponding configuration parameter descriptions, see Table 2 -3. Table 2-4, "Note", Figure 2-7, Figure 2-8, Figure 2-9; 4. For updated software version requirements, see "Note".	V1.7
2022-10-31	V1.8	1. Add the IP version change record corresponding to the V1.8 version; 2. For updated software version requirements, see "Note"; 3. Update the resource utilization list, see Table 2-6; 4. Adjust the ck delay debugging interface (force_ck_dly_en, force_ck_dly_set_bin) and update the port description, see Table 2-19.	V1.8

2023-01-13	V1.9	1. Add the IP version change record corresponding to the V1.9 version; 2. Update applicable devices and packages, see Table 2-1; 3. For updated software version requirements, see "Note"; 4. Update the resource utilization list, see Table 2-6; 5. Added dbg_error_status signal description, see "2.5.52 debug_calib_ctrl Description"; 6. Added SCBV setting description chapter, see "2.8.6 SCBV Settings"; 7. Updated reference documents, see "3.1 Reference Documents".	V1.9
2023-04-28	V1.10	1. Add the IP version change record corresponding to the V1.10 version; 2. For updated software version requirements, see "Note"; 3. For UI interface optimization, update Memory Type and device type related configurations, see Table 2-2 and Table 2-3; 4. Update the resource utilization list, see Table 2-6; 5. Modify the description of AXI address mapping relationship, see Figure 2-16; 6. Update the debug_data field definition and add debugging ports (this_group_ca_dly, ck_dqs_diff), see Table 2-22; 7. Improve the debug_calib_ctrl field definition and update the debug port dbg_error_status description, see Table 2-23; 8. Update the description of the key indicator signal port err_flag_led, see Table 2-33.	V1.10
2023-06- 28	V1.11	1. Add the IP version change record corresponding to the V1.11 version; 2. Update applicable devices and packages, see Table 2-1; 3. For updated software version requirements, see "Note"; 4. Update the resource utilization list, see Table 2-6; 5. Delete some redundant debug ports, see Table 2-22; 6. Update the cover and file naming according to the naming convention.	V1.11
2023-08- 31	V1.12	1. Add the IP version change record corresponding to the V1.12 version; 2. 2. Update module instantiation and add Memory Address Mapping Selection option description, see Table 2-2; 3. Update the resource utilization list, see Table 2-6; 4. Update the AXI4 interface description, see "2.5.2 1 Simplified AXI4 Interface"; 5. Update the dbg_slice_status description, see Table 2-24.	V1.12

2023-10-31	V1.13	1. Add the IP version change record corresponding to the V1.13 version; 2. Update applicable devices and packages, see Table 2-1; 3. For updated software version requirements, see "Note"; 4. Modify the default values of the Memory Part option and Select Base Part option on the Memory Options page, see Table 2-3; 5. Update the resource utilization list, see Table 2-6; 6. Modify the DFI interface write operation timing diagram, see Figure 2-25; 7. Added instructions for using corresponding clock constraints with different synthesis tools, see "Note".	V1.13
2023-11-28	V1.13	1. Add the IP version change record corresponding to V1.13a version; 2. Update the "Document Version Revision Record" and improve the "Applicable IP and Corresponding Version" and "Date".	V1.13, V1.13a
2024-03-28	V1.13b	1. Add the IP version change record corresponding to V1.13b version; 2. Update the "Document Version Revision Record" and improve the "Applicable IP and Corresponding Version" and "Date"; 3. Update applicable devices and packages, see Table 2-1; 4. For updated software version requirements, see "Note"; 5. Update the resource utilization list, see Table 2-6.	V1.13b

IP version change history

Version	Release Notes	Release date
V1.0	1. PG2L HMIC_S IP initial version.	2020-04-30
V1.1	1. PG2L HMIC_S IP iterative version to avoid hardware bugs.	2020-05-27
V1.2	1. PG2L HMIC_S IP iterative version to avoid hardware bugs.	2020-06-08
V1.3	1. The reference design is complete; 2. The debug function is perfect; 3. Fixed the problem of occasional cpd not locking after multiple resets; 4. Optimize the timing of dll update; 5. Clock scheme optimization; 6. PLL configuration parameter optimization; 7. cdc problem fixed; 8. The GUI interface adds configurable AL, CL, and CWL parameters; 9. Join the vcs simulation environment.	2020-09-30
V1.4	1. Support PG2L100H multi-package.	2020-10-20
V1.5	1. The rcf constraint is added by default in the reference design project. If the customer builds his own project, he needs to add the rcf constraint according to the UG instructions; 2. Support PLL reference clock constraints to banks other than CA; 3. Clock constraint optimization; 4. Optimization of reading calibration and eye diagram calibration functions; 5. Turn off regular dll updates by default; 6. Fixed the jump problem of wl_p_dll_bin during write calibration; 7. Enhance bist function.	2020-11-13
V1.5a	1. Improve the stability of 1066Mbps multiple resets; 2. Solve the 800x48/800x64 error code problem; 3. Fixed the error in the axi interface packet length description in UG. NOTE: This version is only for PG2L100H.	2020-12-12
V1.6	1. Added PG2T330H HRIO FFBG900 and FFBG676 package support; 2. Optimization of GPLL and PPLL clock configuration parameters, clock constraints, etc.; 3. Fixed the error in the axi interface packet length description in UG; 4. Fix some bugs and improve stability. Note: This version is only for PG2T330H.	2020-12-29
V1.6a	1. Modify the device name, PG2T330H is updated to PG2T390H.	2021-07-20
V1.6b	1. Fix the bug of incorrect DQS pin assignment and some other constraint issues; 2. Change the PDS version number in the IP, from "1.0.3" to "1.0.5"; 3. Solved the problem of GTP ISERDES CLKB port in simulation; 4. Added ck delay top-level configurable port, the default setting is 5 steps; 5. Optimize the DQS alignment ck judgment code in the wrlvling stage; 6. Update UG, adjust the manual structure, and standardize the format; 7. Fix the bug of idf file reload error; 8. The GUI interface is optimized, the default CA_GROUP selected interface option is deleted, the FDC import function is added, and the description of the reference voltage and particle capacity is added to the summary page.	2021-12-03

V1.7	1. Add support for PG2L25H and PG2L50H devices; 2. The GUI interface is optimized, the SDRAM particle model is updated, the enable switch option of the cs_n signal is added to the interface, and the CA signal is checked for legitimacy; 3. Fix critical warnings Route-2036; 4. Controller code upgrade; 5. UG update.	2022-04-29
V1.8	1. Fixed the problem of incorrect cpd position constraint of the right BANK; 2. Fix the problem of incorrect data bit width in idf reload; 3. When adapting write leveling, the initial delay of ck is smaller than the initial delay of dqs; 4. UG update.	2022-10-31
V1.9	1. Add PG2L200H device support; 2. Add PG2L100HX device support; 3. Modify the phy_dq_sysclk clock constraint mode to edge alignment; 4. Modify the slice_top module name to change with the project; 5. Add dbg_error_status port; 6. UG update.	2023-01-13
V1.10	1. Improve the write leveling module to support a larger training range; 2. Bist upgrade; 3. Print log information during the simulation process; 4. Fix the feedback clock error of some PLLs; 5. Fixed the problem of row address and bank address conversion errors in the controller code; 6. Generate different simulation models for different particle widths; 7. Remove DDR2 and LPDDR in the IP configuration interface; 8. UG update.	2023-04-28
V1.11	1. Adapt to PG2K400 HRIO; 2. Fixed the constraint error of PG2T390's default configuration of 50M clock; 3. Fixed the issue of GATE CLMA constraint position error under special configuration; 4. Fixed the problem of duplication in automatic configuration of DQ pins; 5. Modify some code risks detected by spyglass; 6. Fixed the problem that the generated port file does not contain ref_clk; 7. UG update.	2023-06-28
V1.12	1. Serialization of the training process; 2. Optimize bus port; 3. Instantiate different codes for single PHY and controller+phy options; 4. Solve the risk of overflow in the maximum number of steps in the wrcal process; 5. Fix the problem that idelay is not retained in the rdcal process; 6. Adjust the clock of the ca signal in slice_top; 7. Solve problems found during the verification process; 8. Add the option to adjust the order of row bank col in the configuration interface; 9. UG update.	2023-08-31
V1.13	1. Adapted to PG2T70H; 2. The default particle option is changed to MT41K256M16XX; 3. Modify the fdc file, add OEM constraints, and default to comment state; 4. Fixed the problem that when some banks only have 3 groups, non-existent groups can be selected, resulting in error reports; 5. Modify the mem_rst signal so that any pin can be selected; 6. UG update	2023-10-31
V1.13a	1. PG2L100H FBG484 package supports x64 bit width.	2023-11-28
V1.13b	1. Adapted to PG2T390HX; 2. UG update.	2024-03-28

Table of contents

List of tables

List of figures

Chapter 1 Preface

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

1.1 About this manual

This manual is the user guide for HMIC_S (High performance Memory Interface Controller Soft core) IP, a DDR3 IP launched by Unisoc. The contents of this manual mainly include IP usage guidelines and related appendices. Through this manual, users can quickly understand the related features and usage of HMIC_S IP.

1.2 Manual writing standards

Table 1-1 Writing specifications

Word	Usage principles
Notice	If the user neglects to pay attention to the content, there may be certain adverse consequences due to misoperation or the operation may not be successful.
Description	Instructions and tips provided to users.
Recommend	Recommended settings and usage instructions for users.

Chapter 2 IP user's guidance

This chapter describes the HMIC_S IP related usage guide, including IP introduction, IP block diagram, IP generation process, Example Design, IP interface description, IP register description, typical applications, instructions and precautions, and IP debugging methods. For more detailed information about the design process, please refer to the following PDS help document.

- *Pango_Design_Suite_Quick_Start_Tutorial* [1]
- *Pango_Design_Suite_User_Guide* [2]
- *IP_Compiler_User_Guide* [3]
- *Simulation_User_Guide* [4]

2.1 IP Introduction

HMIC_S IP is a DDR3 IP launched by Unisoc, which is compatible with LPDDR and DDR2. The configuration and generation of IP modules can be completed through the IPC (IP Compiler) tool in the company's PDS (Pango Design Suite) suite.

2.1.1 Main features

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

- Support DDR3, DDR2, LPDDR;
- Supports maximum data bit width 72bit;
- User interface: simplified AXI4 bus interface, APB bus interface;
- Supports configurable low-power modes: Self-Refresh and Power Down;
- Supports DDR3 with a maximum data rate of 1066Mbps;
- Supports DDR2’s maximum data rate up to 800Mbps;
- Supports the maximum data rate of LPDDR up to 400Mbps;
- Burst Length 8 and single Rank;
- PHY can be used alone.

2.1.1 Applicable devices and packages

Table 2-1 HMIC_S IP Applicable devices and packages

Applicable devices	Support packaging type
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

2.2 IP block diagram

The HMIC_S IP system block diagram is shown in Figure 2-1. HMIC_S IP provides two modes: Controller + PHY and PHY Only.

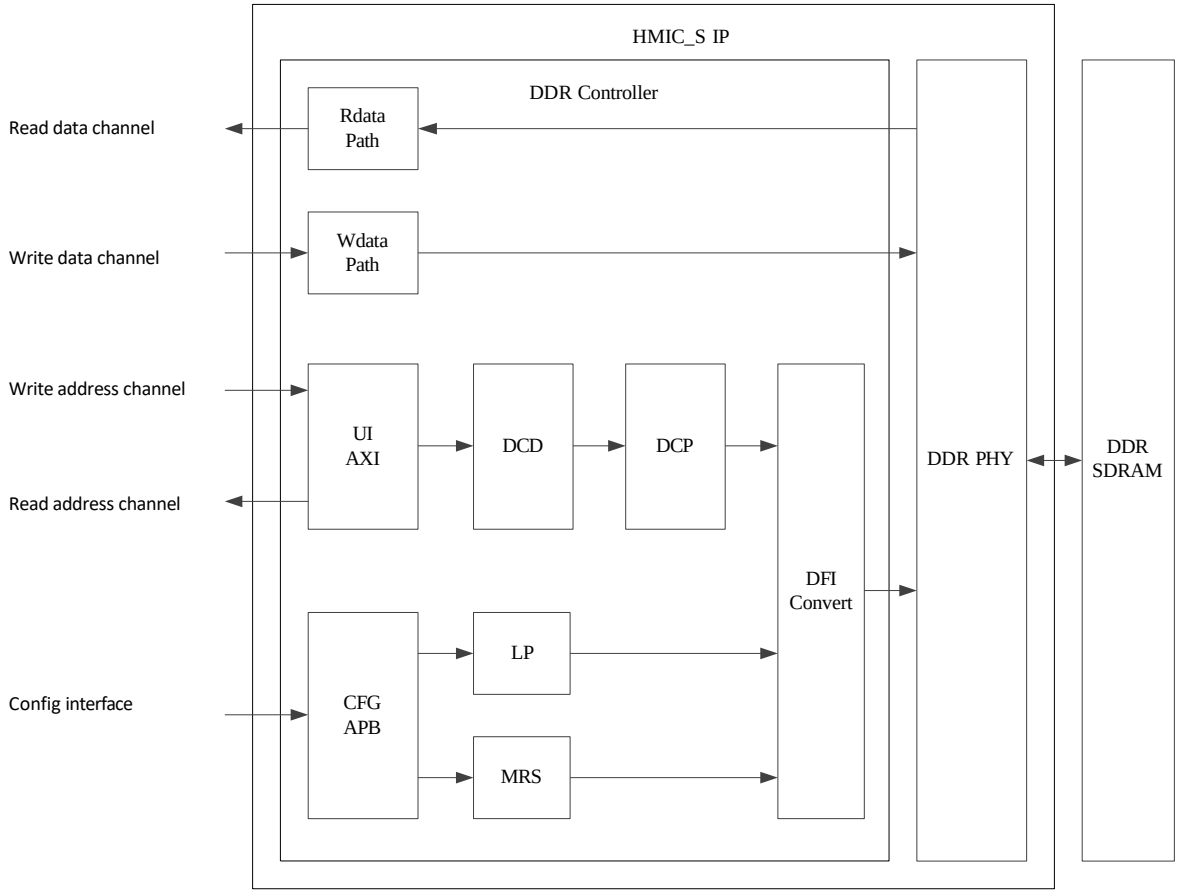


FIGURE 2-1 HMIC_S IP SYSTEM BLOCK DIAGRAM

2.2.3 Controller + PHY

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

- Simplified AXI4 interface

The interface includes four parts: write address channel, read address channel, write data channel and read data channel.

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

The write data is passed directly to the DDR PHY through the write data channel interface, through the Wdata Path module, and finally to the DDR Memory interface.

After the DDR PHY samples and parses the read data from DDR Memory, it is synchronized by the Rdata Path module and returned to the user through the read data channel interface.

- Config interface

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

2.2.2 PHY only

In this mode, users need to implement the Controller themselves design, and through the DFI interface and PHY Docking to realize the operation of DDR SDRAM.

2.3 IP Generate process

2.3.1 Module instantiation

The customized configuration of HMIC_S IP can be completed through the IPC tool, and the required IP modules can be instantiated and generated. For specific usage of IPC tools, please refer to IP_Compiler_User_Guide[3].

The main steps of HMIC_S IP module instantiation are described below.

1. Select IP

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

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

Notice:

- PG2L25H , PG2L50H : The software must use 2022.1 and above;
- PG2L100H , PG2T390H : It is recommended to use software 2021.1-SP7.6 and above;
- PG2L200H : Software must use 2022.2-SP1.2 and above;
- PG2L100HX : Software must use 2023.1 and above;
- PG2K400 : Software must use 2023.2 and above;
- PG2T70H : Software must use 2023.2 and above;
- PG2T390HX : Software recommended to use 2022.2-SP6.4 and above versions.

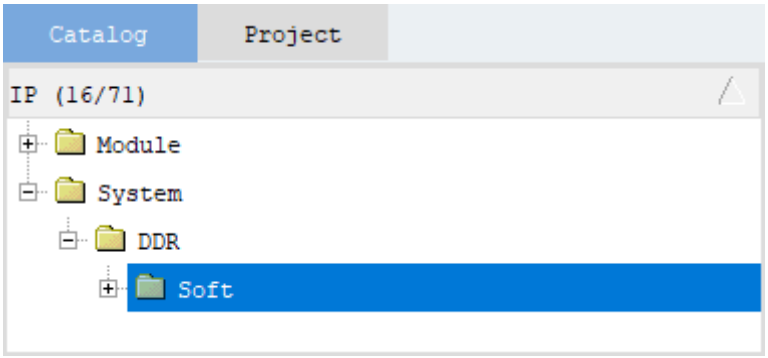


FIGURE 2-2 HMIC_S IP SELECT PATH INTERFACE

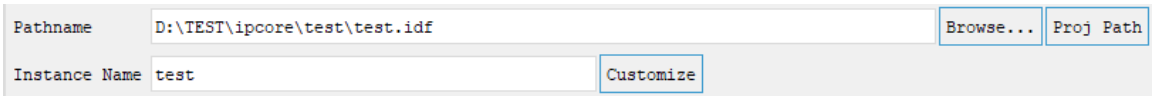


FIGURE 2-3 PROJECT INSTANTIATION INTERFACE

2. Configure IP parameter

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

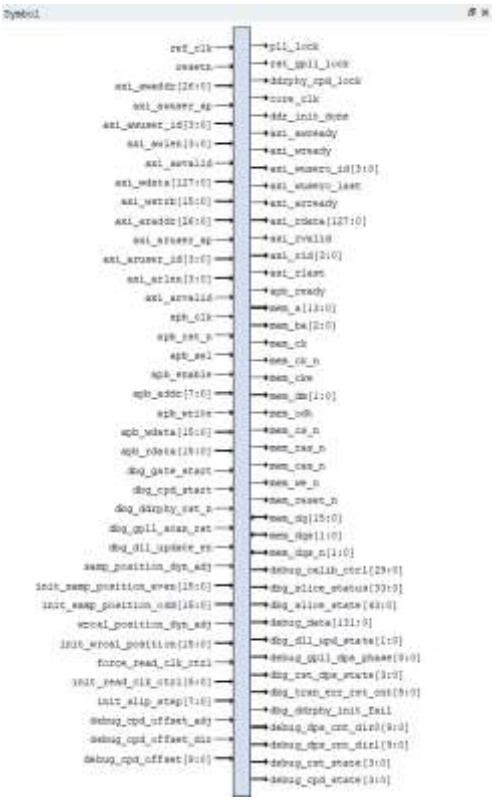


FIGURE 2-4 HMIC_S IP INTERFACE DIAGRAM

FIGURE 2-5 HMIC_S IP CONFIGURATION PARAMETER INTERFACE

Notice:

Please be sure to configure the IP in the order of the page Parameters, i.e. Step 1 → Step 2 → Step 3 → Step 4.

Parameter configuration is divided into four pages, namely Step1: Basic Options, Step2: Memory Options, Step3: Pin/Bank Options, and Step4: Summary. HMIC_S IP configuration steps are described below.

(1) Step 1: Basic Options

Basic Options is the basic configuration page of IP. The interface is shown in Figure 2-6. For parameter description, see Table 2-2.

FIGURE 2-6 BASIC OPTIONS PAGE

Table 2-2 Basic Options Page configuration parameter description

Options area	Option name / parameter name	Parameter Description	Default value
Type Options	Memory Type	The SDRAM type used, the currently available 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 you select Controller + PHY, the generated IP code contains Controller and PHY; If you select PHY Only, the generated IP code contains only the PHY portion.	Controller + PHY
Memory Address Mapping Selection	ROW + BANK + COLUMN	Select the read and write address mapping method of the Controller AXI interface as: "ROW + BANK + COLUMN", see Figure 2-16.	Check
	BANK + ROW + COLUMN	Select the read and write address mapping method of the Controller AXI interface as: "BANK + ROW + COLUMN", see Figure 2-17.	Uncheck
Width Options	Total Data Width [1]	The total DQ width of the off-chip SDRAM connected to HMIC_S. The total widths currently supported are: 1) 72 2) 64 3) 56 4) 48 5) 40 6) 32 7) 24 8) 16 9) 8	16
Clock Settings	Input Clock Frequency	Input clock of HMIC_S, unit MHz.	50
	Desired Data Rate [2]	Desired data rate. DDR3 supports a maximum speed of 1066Mbps. DDR2 supports a maximum speed of 800Mbps. LPDDR supports a maximum speed of 400Mbps.	800Mbps
	Actual Data Rate	The actual achievable data rate is as close as possible to the desired rate.	-
Write and Read Latency	CAS Write Latency (CWL)	CAS Write Latency configuration, unit tCK. (DDR3 only)	5
	CAS Latency (CL)	CAS Latency configuration, unit tCK. (DDR3 only)	6
	Additive Latency(AL)	Additive Latency configuration, unit tCK. (DDR3 only)	CL- 2

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

1 The FBG484 and MBG324 packages of PG2L100H, all packages of PG2L25H and all packages of PG2L50H only support bit width up to 32.
2 The configuration default value of this parameter is related to "Memory Type", DDR3: 800Mbps; DDR2: 600Mbps; LPDDR: 200Mbps.

(2) Step 2: Memory Options

Memory Options is the configuration page for Memory parameters. The interface is shown in Figure 2-7. For parameter description, see Table 2-3.

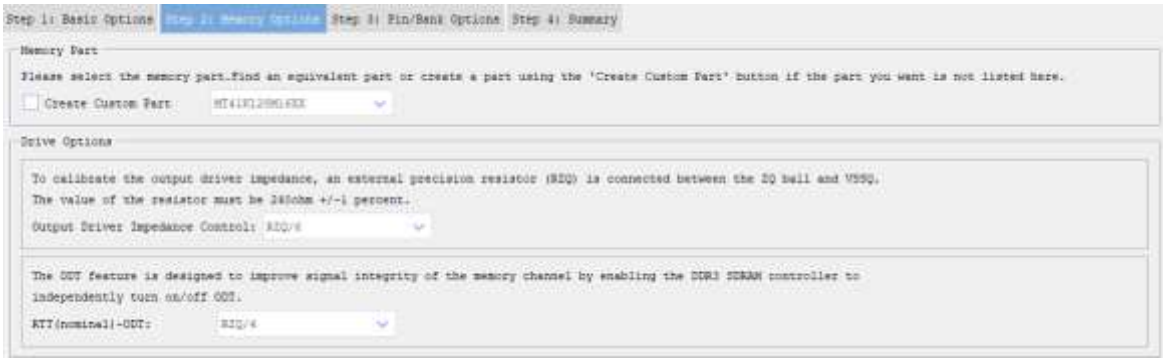


FIGURE 2-7 MEMORY OPTIONS PAGE

Table 2-3 Memory Options Page configuration parameter description

Options area	Option name / parameter name	Parameter Description	Default value
Memory Part	SDRAM device model [3]	The models supported by DDR3 are: 1) MT41K128M8XX 2) MT41K64M16XX 3) MT41K256M8XX 4) MT41K128M16XX 5) MT41K512M8XX 6) MT41K256M16XX	MT41K256M 16XX
	Create Custom Part	Customize new SDRAM type enable selection. If none of the above device models supported by the IP meet the requirements, the user can check this option and customize a new SDRAM type based on the reference model in the Custom Memory Part option box below.	Uncheck
Custom Memory Part [4]	Select Base Part [5]	Custom SDRAM reference model.	MT41K256M 16XX
	Timing Parameters [6]	Custom SDRAM Timing Parameters, including: trfc , tras , trp , trcd , twr , trefi , trtp , twtr .	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
	Row Address [7]	Line address.	14
	Column Address [8]	Column address.	10
	Bank Address [9]	Bank address.	3

Drive Options [10]	Output Driver Impedance Control	DDR3 supported drive capability options. For detailed description, please see: JESD79-3D, DDR3 SDRAM Standard[8].	RZQ/6
	RTT(nominal) -ODT		RZQ/4

3 The configuration default value of this parameter is related to "Memory Type",
DDR3: MT41K128M16XX;
DDR2: MT47H128M16XX-25E;
LPDDR: MT46H128M16XXXX-5L-IT.
4 This option box is displayed when the "Create Custom Part" option is checked and is used to customize a new SDRAM type.
5 The configuration default value of this parameter is related to "Memory Type",
DDR3: MT41K128M16XX;
DDR2: MT47H128M16XX-25E;
LPDDR: MT46H128M16XXXX-5L-IT.
6 The configuration default value of this parameter is related to "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.
7 The configuration default value of this parameter is related to "Memory Type",
DDR3: 14;
DDR2: 14;
LPDDR: 14.
8 The configuration default value of this parameter is related to "Memory Type",
DDR3: 10;
DDR2: 10;
LPDDR: 11.
9 The configuration default value of this parameter is related to "Memory Type",
DDR3: 3;
DDR2: 3;
LPDDR: 2.
10 The drive capability related option parameters and their default configuration are related to “Memory Type”. The drive capability options and their default configuration corresponding to different SDRAM types 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).

(3) Step 3: Pin/Bank Options

Pin/Bank Options is the configuration page for interface parameters. The interface is shown in Figure 2-8. Detailed parameter descriptions are shown in Table 2-4.

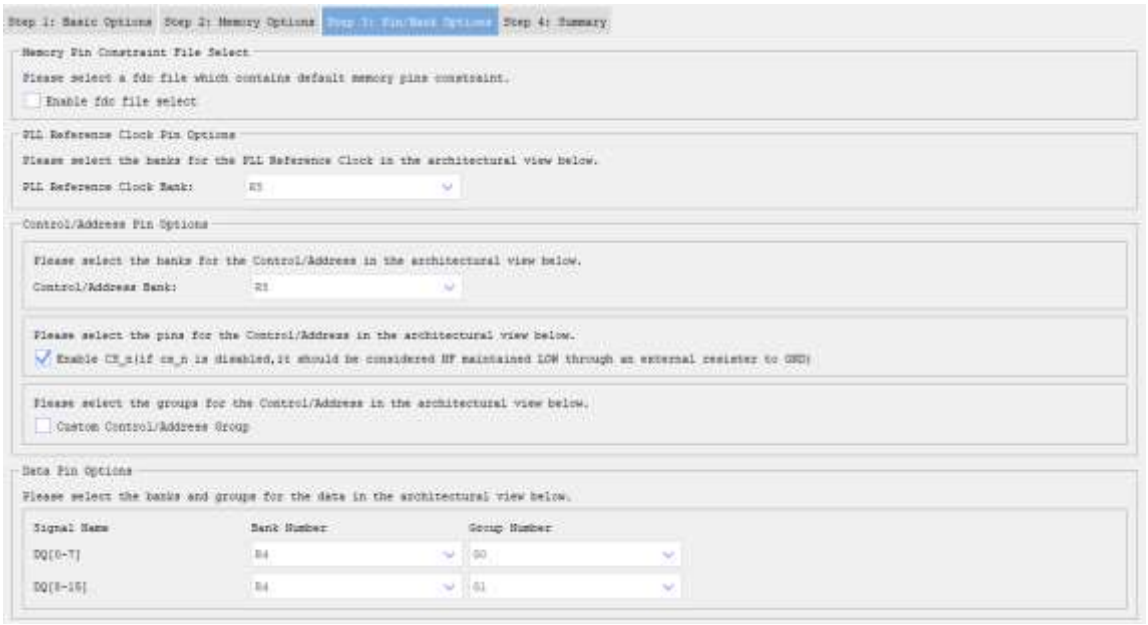


FIGURE 2-8 PIN/BANK OPTIONS PAGE

Notice:

All configuration items in "Step 3: Pin/Bank Options" must be configured according to the pin assignments on the actual board. After generating the IP, constrain the pin positions of the DQ signal, reset signal and status signal according to the pin assignments on the actual board, otherwise errors may occur when running Flow.

Table 2-4 Pin/Bank Options Page configuration parameter description

Options area	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, you can enter the user fdc file path in the text box, and the Memory interface constraints in the fdc file will be automatically read, and the Control/Address and Data Pin in the UI will be configured. Note: The signal name of the Memory interface in the user fdc must be consistent with the signal name of the Memory interface in the Example Design.	Disable
PLL Reference Clock Pin Options	PLL Reference Bank [11]	Bank where the PLL reference clock resides.	L5
Control/ Address Pin	Control/Address Bank [11]	Bank where the control and address lines of the Memory interface are located.	L5
	Enable CS_n	Mem_cs_n signal enable selection.	Enable

Options	Custom Control/Address Group	User-defined control and address bus group enable selection. Checked: User customizes each PAD Grouping and pin constraints; unchecked: default grouping.	Disable
	Control and address signals	This option box is displayed when the "Custom Control/Address Group" option is checked, and is used to select the Group and Pin where the control and address signals correspond to the PAD.	-
Data Pin Options	Bank Number [12]	Select the Bank where DQ[13] is located.	L6
	Group Number [14]	Select the Group where DQ_{13} is located.	G1

Note: " - " means IP There is no default value for this parameter in the configuration interface.

11 The configuration default value of 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.

12 The configuration default value of this parameter is related to the selected device and package. Taking DQ[0-7] as an example,
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.

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

14 The default configuration value of this parameter is related to the selected device and package. Taking DQ[0-7] as an example,

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.

Notice :

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

- When checking the "Custom Control/Address Group" option for custom pin configuration, be sure not to constrain signals to the same pin. If there are pins constrained to the same position, the UI interface will be marked in red to indicate the corresponding Pin Number.

---

(4) Step 4: Summary

The Summary page is used to print the current configuration information and does not require configuration parameters. The page is shown in Figure 2-9.

FIGURE 2-9 SUMMARY PAGE

3. Generate IP

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

FIGURE 2-10 HMIC_S IP GENERATE REPORT INTERFACE

Notice:

The .pds files and .fdc files generated by the IP are for reference only. Please change the pin constraints according to the actual pin connections when using them.

After the IP is successfully generated, the file shown in Table 2-5 will be output under the Project path specified in Figure 2-3.

Table 2-5 IP Generated output file

Output file [15]	Description
$instname.v	The top-level .v file for the Controller + PHY of the generated IP.
$instname_ddrphy_top.v	PHY top-level .v file for the generated IP.
$instname.idf	Configuration file for the generated IP.
/rtl/*	RTL code file for the generated IP.
/example_design/*	Generated IP Example Design Test used Bench And the corresponding Memory Simulation Model document.
/pnr/*	The generated IP Example Design corresponding project file .pds and pin constraint file .fdc.
/sim/*	Simulation directory for generated IP. sim.tcl is the modelsim simulation script, makefile is the vcs simulation script, and sim_file_list.f is the simulation file list.
/sim_lib/*	Directory of encrypted files required by IP.
/rev_1	The default output path of comprehensive reports. (This folder will only be generated after specifying the synthesis tool)
readme.txt	readme file, describing the structure of the generated directory after IP generation.

15) $instname is the instantiated name entered by the user; "*" is a wildcard character, replacing files of the same type.

2.3.2 constraint configuration

For specific configuration methods of constraint files, you can consult the relevant help documents under the PDS installation path: User_Constraint_Editor_User_Guide[5], Physical_Constraint_Editor_User_Guide[6], Route_Constraint_Editor_User_Guide[7].

2.3.3 Run simulation

The simulation of HMIC_S IP is based on the Test Bench of Example Design. See "2.4 Example Design" for details about Example Design.

For more detailed information about PDS simulation functions and third-party simulation tools, you can consult the relevant help documents under the PDS installation path: Pango_Design_Suite_User Guide[2], Simulation_User_Guide[4].

1. Modelsim simulation

Simulation running steps: Open cmd.exe, switch the current directory to /sim in the IP generation directory in the command line, run vsim, and open the ModelSim simulation software, as shown in Figure 2-11.

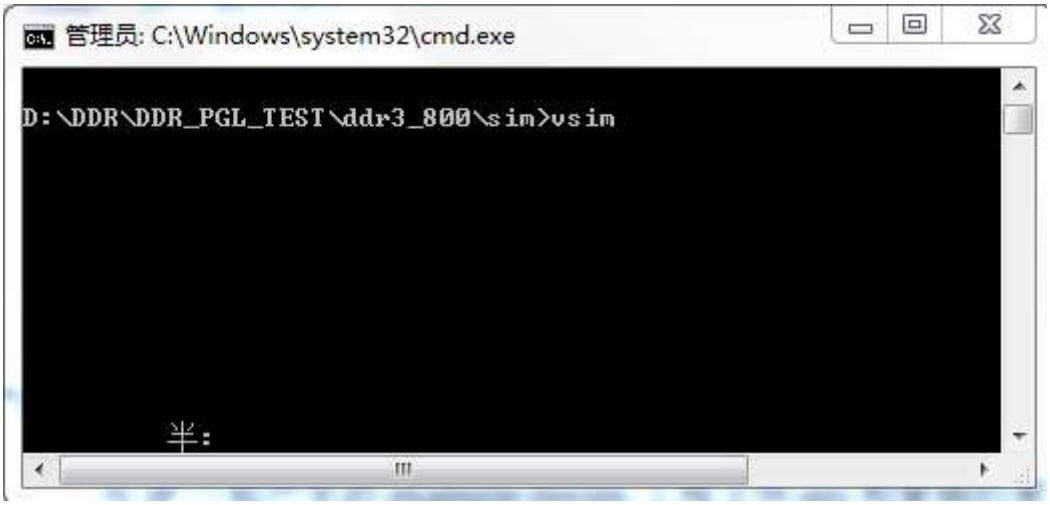


FIGURE 2-11 OPEN MODELSIM INSTRUCTION

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

FIGURE 2-12 MODELSIM EXECUTE SIMULATION SCRIPT

2. VCS simulation

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

FIGURE 2-13 VCS EXECUTE SIMULATION SCRIPT

2.3.4 Synthesis and place and route

For specific usage of the PDS synthesis tool and placement and routing tool, please refer to the help document under the PDS installation path.

Notice:

The Example Design project file .pds and pin constraint file .fdc generated together with the IP are stored in the /pnr/example_design directory. The physical constraints need to be modified according to the actual device used and the routing of the PCB board. For details, please refer to "2.8 Description and Precautions".

2.3.5 Resource utilization

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

Device	Config mode	IP Operating mode	Resource utilization typical value				
			LUT	FF	GPLL	PPLL	USCM
PG2L25H	DDR3x32	Controller + PHY	4208	4094	2	2	2
PG2L50H	DDR3x32	Controller + PHY	4200	4094	2	2	2
PG2L100H	DDR3x32	Controller + PHY	4711	4179	2	2	2
	DDR3 x64	Controller + PHY	6885	5794	2	3	2
PG2L100HX	DDR3x32	Controller + PHY	4290	4148	2	2	2
	DDR3 x64	Controller + PHY	6479	5732	2	3	2
PG2L200H	DDR3x32	Controller + PHY	4209	4094	2	2	2
	DDR3 x64	Controller + PHY	6274	5676	2	3	2
PG2T390H	DDR3x16	Controller + PHY	3528	3373	2	2	2
	DDR3x32	Controller + PHY	4703	4183	2	2	2
PG2K400	DDR3x32	Controller + PHY	4260	4150	2	2	2
	DDR3 x64	Controller + PHY	6356	5734	2	3	2
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

2.4 Example Design

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

2.4.1 Design block diagram

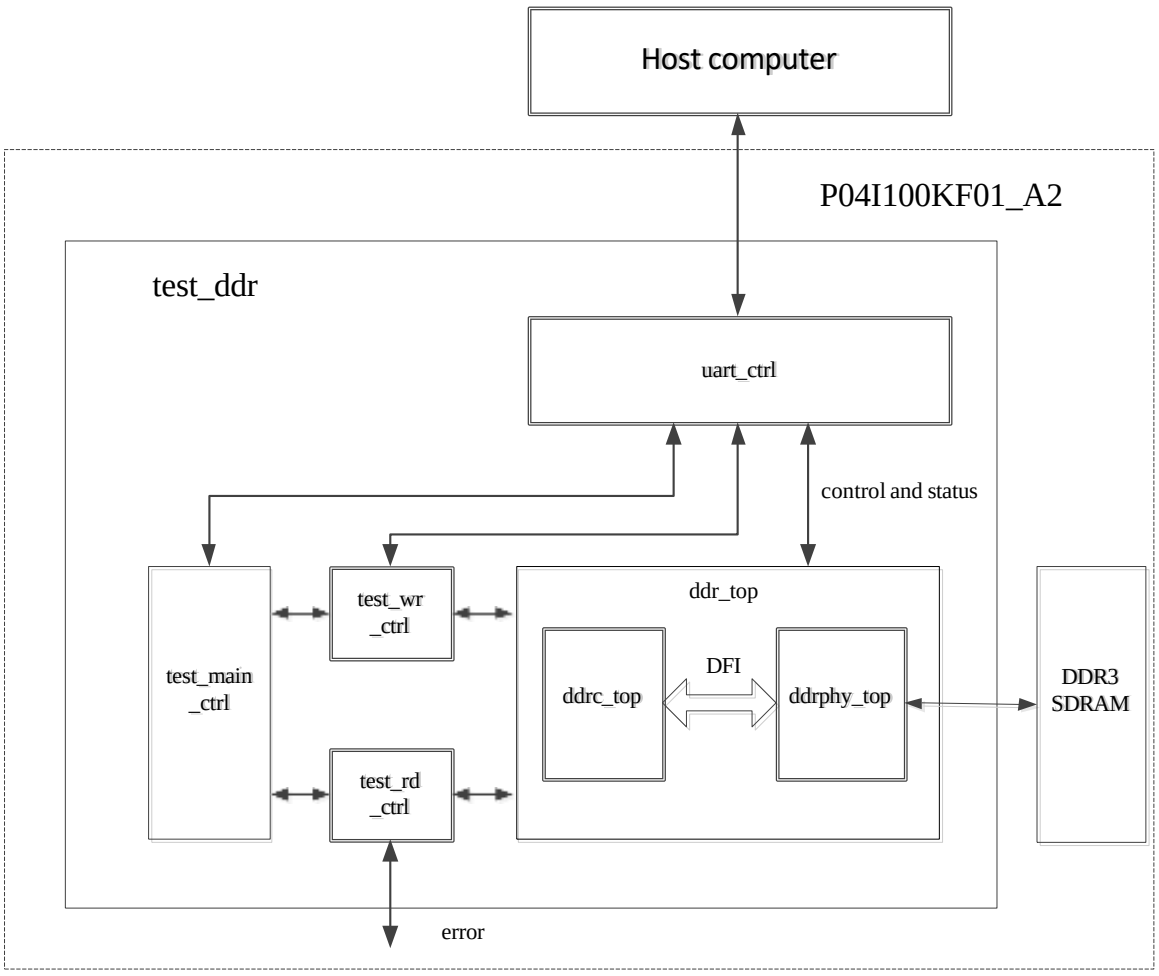


FIGURE 2-14 EXAMPLE DESIGN SYSTEM BLOCK DIAGRAM

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

2.4.2 Test Methods

In the Example Design, the user logic reads and writes the HMIC_S IP and verifies the readback data. The detailed test process is shown in Figure 2-15.

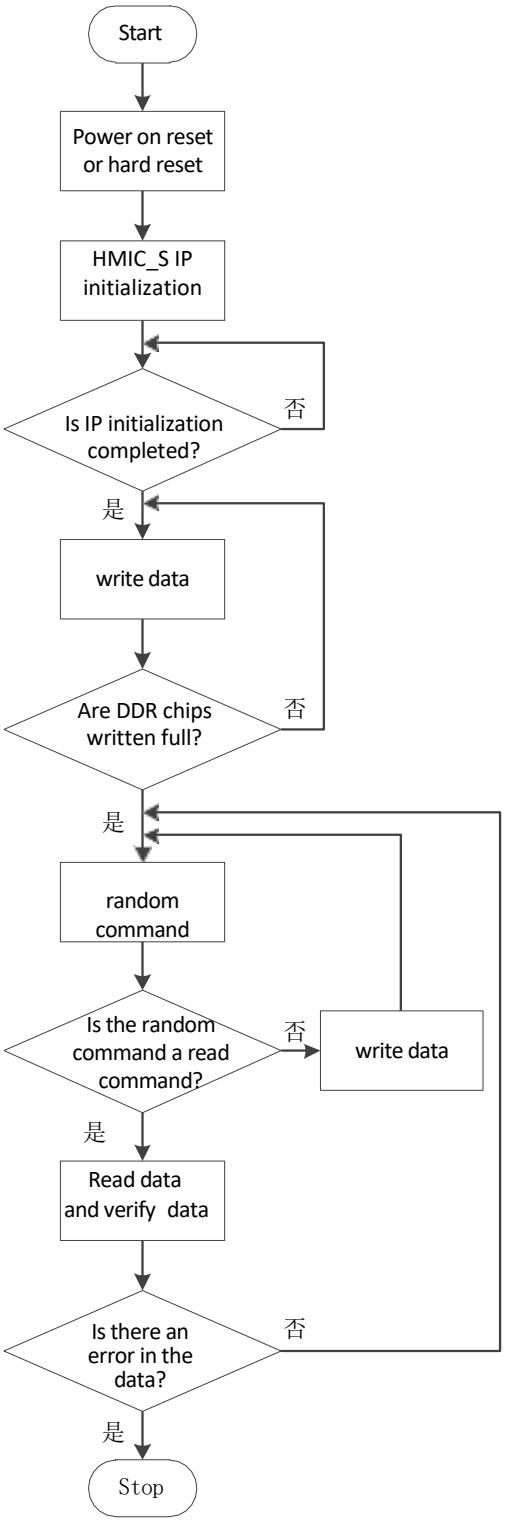


FIGURE 2-15 EXAMPLE DESIGN TEST FLOW CHART

After the system is powered on or started with a hard reset, the HMIC_S IP starts to perform initialization. After the initialization is completed (ddrc_init_done is pulled up), the test_main_ctrl module controls the test_wr_ctrl module to generate write instructions and write data to initialize the data of the DDR particles. After the writing is full, test_main_ctrl starts random Read and write, test_rd_ctrl checks the data read back to determine whether the data is wrong.

Notice:

The Example Design generated with IP cannot directly run Flow on-board testing. It is necessary to constrain the pins according to the actual pin connection relationship of the board, and then run Flow on-board testing.

2.5 IP Interface description

This section introduces the HMIC_S IP related interface description and timing description.

2.5.1 Controller Interface Description

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 level	External reset input.
phy_init_done	I	1	high level	ddrphy The initialization completion flag: 1'b1 : ddrphy Initialization completed; 1'b0 : ddrphy Initialization is not completed;
ddr_init_done	O	1	high level	IP initialization completion flag: 1’b1: ddr IP initialization has been completed; 1’b0: ddr IP initialization is not completed, and external operations on ddr IP are invalid.

Note: " - " means there is no such parameter.

2. Simplified AXI4 interface

(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 the address.
axi_awuser_ap	I	1	high level	AXI Write and automatically precharge .
axi_awuser_id	I	4	-	AXI Write address ID .
axi_awlen	I	4	-	AXI Write burst length.
axi_awready	O	1	high level	AXI Write address ready .
axi_awvalid	I	1	high level	AXI Write address valid .

Note: " - " means there is no such parameter.

(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 level	AXI Read and automatically precharge .
axi_aruser_id	I	4	-	AXI Read address ID.
axi_arlen	I	4	-	AXI Read burst length.
axi_arready	O	1	high level	AXI Read address ready.
axi_arvalid	I	1	high level	AXI read addressvalid

Note: " - " means there is no such parameter.

(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 level	AXI Write data strobes .
axi_wready	O	1	high level	AXI Write data ready .
axi_wusero_id	O	4	-	AXI Write data ID .
axi_wusero_last	O	1	high level	AXI Write data last .

Note: " - " means there is no such parameter.

(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 level	AXI Read data last Signal.
axi_rvalid	O	1	high level	AXI Read data valid .
axi_rdata	O	DQ_WIDTH*8	-	AXI Read data.

Note: " - " means there is no such parameter.

3. Config interface

Table 2-12 Config interface

Port	I/O	Bit width	Valid values	Description
apb_clk	I	1	high level	APB clock.
apb_rst_n	I	1	low level	APB reset.
apb_sel	I	1	high level	APB Select .
apb_enable	I	1	high level	APB port enable .
apb_addr	I	8	-	APB address bus.
apb_write	I	1	high level	APB Reading and writing direction, high level writing, low level reading.
apb_ready	O	1	high level	APB PortReady .
apb_wdata	I	16	-	APB Write data.
apb_rdata	O	16	-	APB Read data.

Note: " - " means there is no such parameter.

4. DFI interface

Table 2-13 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 level	The read data of the dfi interface is valid.

dfi_rddata	I	DQ_WIDTH*8	-	DFI interface data.
dfi_wrdata_en	O	4	high level	DFI write enable signal .
dfi_wrdata	O	DQ_WIDTH*8	-	DFI Write data.
dfi_wrdata_mask	O	DQ_WIDTH*8/8	high level	DFI Write data byte mask .
dfi_init_complete	I	1	high level	PHY has completed training related operations and is in normal state.
dfi_error	I	1	high level	PHY training Error indication.
dfi_phyupd_req	I	1	-	PHY Request an update .
dfi_phyupd_ack	O	1	-	The feedback signal of dfi_phy_req allows the PHY to update.

Note: " - " means there is no such parameter.

2.5.2 Controller Interface timing description

The Simplified AXI4 interface uses a tailored AXI4 protocol. The Config interface uses the APB protocol.

1. Simplified AXI4 interface

The differences between the Simplified AXI4 interface defined in this design and the standard AXI4 protocol are shown in Table 2-14 and Table 2-15.

Table 2-14 Simplified AXI4 with standard AXI4 difference

Aisle	difference
Write address channel	Reserved AWID, AWADDR, AWLEN, AWUSER, AWVALID, AWREADY end port; Remove AWSIZE, AWBURST, AWLOCK, AWCACHE, AWPROT, AWQOS, and AWREGION ports.
Write data channel	Reserve WID , WDATA , WSTRB , WLAST , WREADY Port; Remove WUSER and WVALID port.
Write responsechannel	N/A .
Read address channel	Reserved ARID , ARADDR , ARLEN , ARUSER , ARVALID , ARREADY port; Remove ARSIZE , ARBURST , ARLOCK , ARCACHE , ARPROT , ARQOS , ARREGION ports.
Read data channel	Keep RID , RDATA , RLAST , RVALID ; remove RRESP , RUSER port.
Low-power interface	N/A .
Clock	AXI with MC clock same.

Table 2-15 Simplified AXI4 with standard AXI4 Detailed differences

Interface signal Classification	Standard AXI4	Simplifying AXI4	signal source	Description	Compatible methods
Global signals	ACLK	clk	Clock Source	Global clock signal	PHY core divided clock
	ARESETn	rst_n	Reset Source	Global reset signal, active low	MC reset clock
Write address channel signals	AWID[16]	axi_awuser_id	Master	Write address ID	Fixed value can be assigned when not in use
	AWADDR[16]	axi_awaddr	Master	write address	See footnote
	AWLEN [16]	axi_awlen	Master	Burst write length	Default burst size is 8
	AWSIZE [17]	N/A	Master	Burst write size	See footnote
	AWBURST [17]	N/A	Master	Burst type	See footnote
	AWLOCK [17]	N/A	Master	lock type	See footnote
	AWCACHE [17]	N/A	Master	cache type	See footnote
	AWPROT [17]	N/A	Master	protection type	See footnote
	AWQOS [17]	N/A	Master	Write QOS identifier	See footnote
	AWREGION [17]	N/A	Master	write domain identifier	See footnote
	AWUSER [16]	axi_awuser_ap	Master	User-defined, written and automatically precharged	See footnote
	AWVALID [16]	axi_awvalid	Master	The write address is valid	See footnote
	AWREADY [16]	axi_awready	Slave	Prepare to write address	See footnote
Write data channel signals	WID [16]	axi_wusero_id	Master	Write ID	Fixed value can be assigned when not in use
	WDATA [16]	axi_wdata	Master	write data	See footnote
	WSTRB [16]	axi_wstrb	Master	write strobe	See footnote
	WLAST [16]	axi_wusero_last	Master	write last mark	See footnote
	WUSER [17]	N/A	Master	Custom	See footnote
	WVALID [17]	N/A	Master	write valid	See footnote
	WREADY [16]	axi_wready	Slave	preparation for writing	See footnote
Write response channel signals	BID [18, 19]	N/A	Slave	Response ID	See footnote
	BRESP [18, 19]	N/A	Slave	write response	See footnote
	BUSER [18, 19]	N/A	Slave	Custom	See footnote
	BVALID [18, 19]	N/A	Slave	Write response is valid	See footnote
	BREADY [17, 19]	N/A	Master	Write response	See footnote

				preparation	
Read address channel signals	ARID [16]	axi_aruser_id	Master	Read ID	Fixed value can be assigned when not in use
	ARADDR [16]	axi_araddr	Master	Read data address	See footnote
	ARLEN [16]	axi_arlen	Master	burst length	See footnote
	ARSIZE [17]	N/A	Master	burst size	See footnote
	ARBURST [17]	N/A	Master	Burst type	See footnote
	ARLOCK [17]	N/A	Master	lock type	See footnote
	ARCACHE [17]	N/A	Master	cache type	See footnote
	ARPROT [17]	N/A	Master	Contains type	See footnote
	ARQOS [17]	N/A	Master	Read address QOS identifier	See footnote
	ARREGION [17]	N/A	Master	Read address field identifier	See footnote
	ARUSER [16]	axi_aruser_ap	Master	User-defined, read and automatically precharge	See footnote
	ARVALID [16]	axi_arvalid	Master	The read address is valid	See footnote
	ARREADY [16]	axi_arready	Slave	Read address preparation	See footnote
Read data channel signals	RID [16]	axi_rid	Slave	Read ID	See footnote
	RDATA [16]	axi_rdata	Slave	Read data	See footnote
	RRESP [18]	N/A	Slave	read response	See footnote
	RLAST [16]	axi_rlast	Slave	Read the last data	See footnote
	RUSER [18]	N/A	Slave	Custom	See footnote
	RVALID [16]	axi_rvalid	Master	Read valid	See footnote
	RREADY [17]	N/A	Master	Reading preparation	See footnote
Low- power interface signals	CSYSREQ [17]	N/A	clock controller	System low power request	See footnote
	CSYSACK [17]	N/A	peripheral equipment	Low power request response	See footnote
	CACTIVE [17]	N/A	peripheral equipment	clock activity	See footnote

16 For the 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", "2.5. 2 1(4) Read Data Channel Timing".
17 Based on the DDR3 IP instantiation module, an encapsulation module layer is provided, and the input interface is reserved as an input signal without being connected to the controller.
18 Based on the DDR3 IP instantiation module, an encapsulation module layer is provided, an output interface is reserved, and a fixed value can be assigned as an output signal.
19 For MC, the write response channel is not provided internally, and it is recommended that the master device ignores the write response channel function.

The Simplified AXI4 interface contains 4 channels, namely write address channel, read address channel, write data channel, and read data channel. Each channel is independent of each other. Generally speaking, the address needs to be sent first, and then the data can be sent and received. Users can use different ID numbers to determine whether the returned data is the requested value.

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

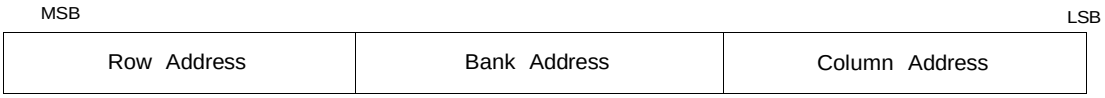


FIGURE 2-16 SIMPLIFIED AXI4 MEMORY OF INTERFACE ADDRESS MAPPING ADDRESS 1

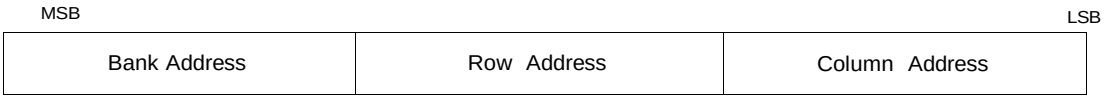


FIGURE 2-17 SIMPLIFIED AXI4 MEMORY OF INTERFACE ADDRESS MAPPING ADDRESS 2

Description:

Column Address Rounding occurs when a user request is on a Column boundary. For related examples, please refer to "2.8.2 Column Address Rounding Examples".

(1) Write address channel timing

The signals included in the write address channel are: axi_awready, axi_awvalid, axi_awaddr, axi_awuser_ap, axi_awuser_id, axi_awlen. Typical timing is shown in Figure 2-18.

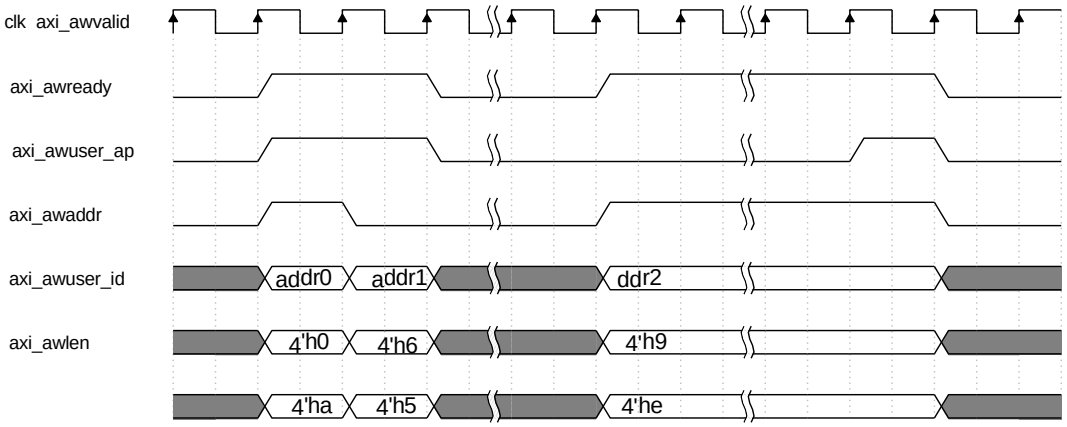


FIGURE 2-18 TYPICAL TIMING OF WRITING ADDRESS

- Conditions for completing a handshake: axi_awready and axi_awvalid are valid at the same time.
- The packet length is controlled by axi_awlen, and the packet length is the value of axi_awlen plus 1.
- Handshake process: Starting from the rising edge of the clock where axi_awvalid is valid, axi_awaddr, axi_awuser_ap, axi_awuser_id, axi_awlen need to remain unchanged until released after the handshake is completed.,

(2) Read address channel timing

The signals included in the read address channel are: axi_arready, axi_arvalid, axi_araddr, axi_aruser_ap, axi_aruser_id, axi_arlen. Typical timing is shown in Figure 2-19.

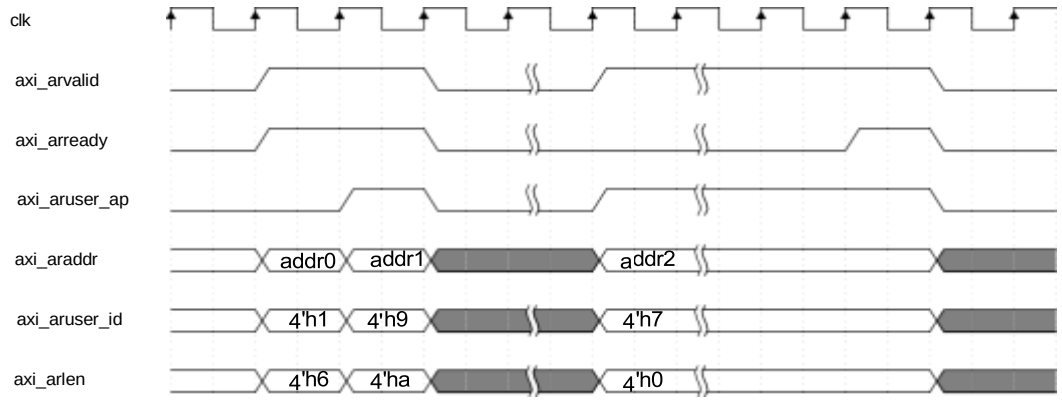


FIGURE 2-19 TYPICAL TIMING OF READING ADDRESS

- Conditions for completing a handshake: axi_arready and axi_arvalid are valid at the same time.
- The packet length is controlled by axi_arlen, and the packet length is the value of

axi_arlen plus 1.

- Handshake process: Starting from the rising edge of the clock where axi_arvalid is valid, axi_araddr, axi_aruser_ap, axi_aruser_id, axi_arlen need to remain unchanged until released after the handshake is completed.

(3) Write data channel timing

The write data channel contains signals: axi_wready, axi_wusero_id, axi_wusero_last, axi_wdata and axi_wstrb. Typical timing is shown in Figure 2-20.

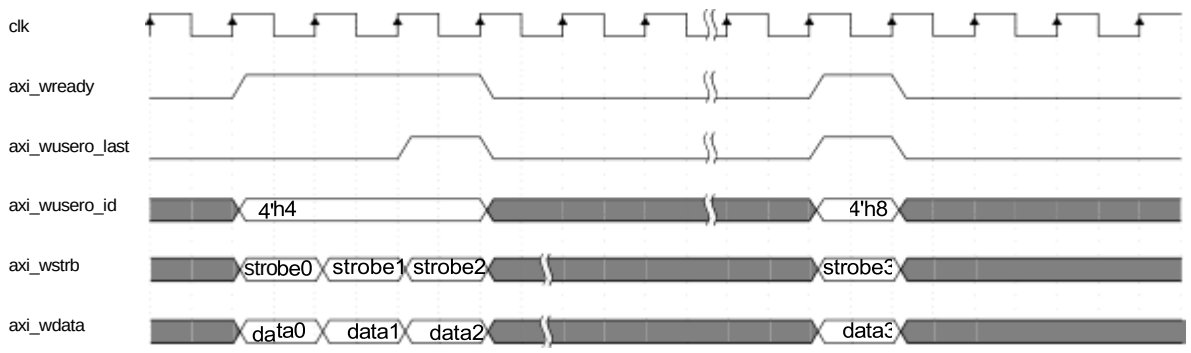


FIGURE 2-20 TYPICAL TIMING OF WRITING DATA

- Transmission valid indication: axi_wready is valid.
- A transmission end indication: axi_wusero_last is valid.
- During transmission: axi_wready, axi_wusero_id and axi_wusero_last are received synchronously, and axi_wdata and axi_wstrb are sent synchronously.

(4) Read data channel timing

Read the signals contained in the data channel: axi_rdata , axi_rid , axi_rlast and axi_rvalid . The typical timing sequence is shown in the figure 2-21 shown.

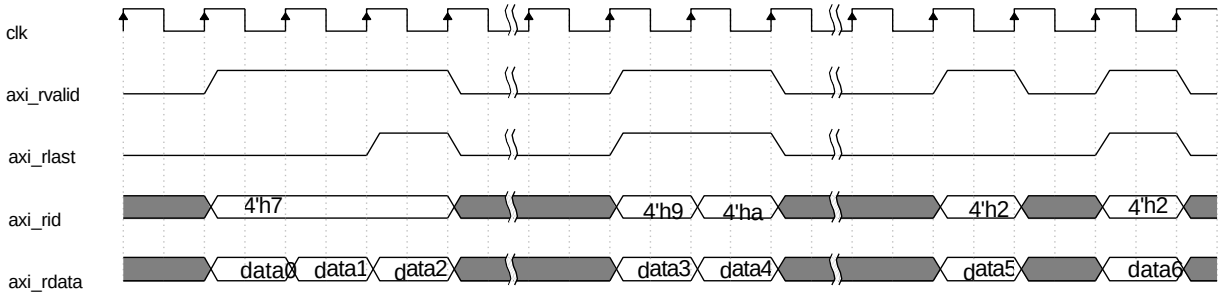


FIGURE 2-21 TYPICAL TIMING OF READING DATA

- Transmission valid indication: axi_rvalid is valid.
- A transmission end indication: axi_rlast is valid.

2. Config interface

The Config interface adopts the standard APB protocol. By configuring the corresponding registers, DDR3 SDRAM can switch between Power Down, Self-Refresh, MRS and Normal states. By reading the corresponding register, the current status of DDR3 can be queried.

The APB interface is a half-duplex communication, with independent read and write data lines, and multiplexed control lines and address lines. Each handshake requires at least 2 apb_clk cycles.

The signals included in the APB interface are: apb_enable, apb_ready, apb_rdata, apb_sel, apb_write, apb_wdata, apb_clk.

Conditions for completing a handshake: apb_enable and apb_ready are valid at the same time.

(1) APB Interface write timing

The typical write timing of the APB interface is shown in Figure 2-22.

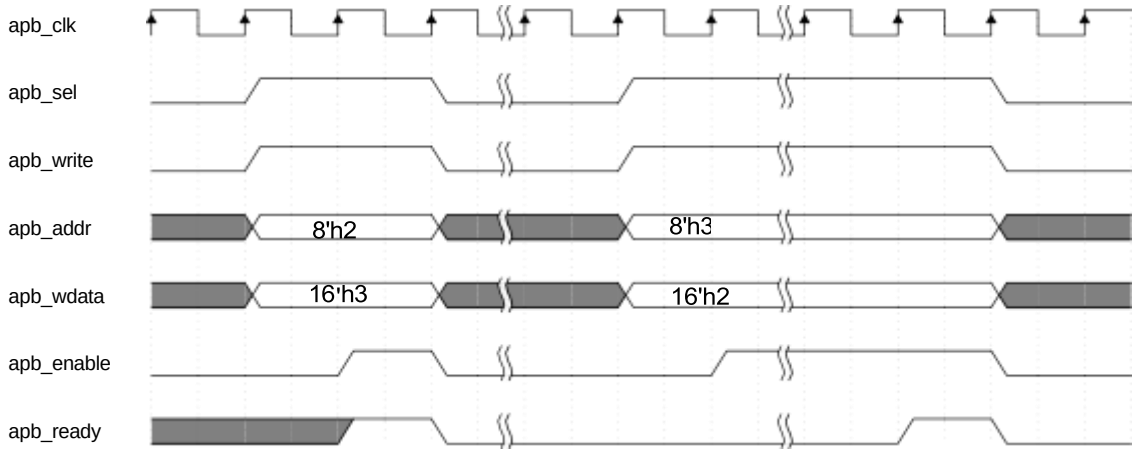


FIGURE 2-22 APB INTERFACE TYPICAL WRITE TIMING

- The first clock cycle: apb_sel, apb_write are pulled high, apb_addr, apb_wdata are given initial values, and they need to remain stable until they are released after the handshake is completed.
- Second clock cycle: apb_enable is pulled high until released after the handshake is completed.

(2) APB Interface read timing

The typical read timing of the APB interface is shown in Figure 2-23.

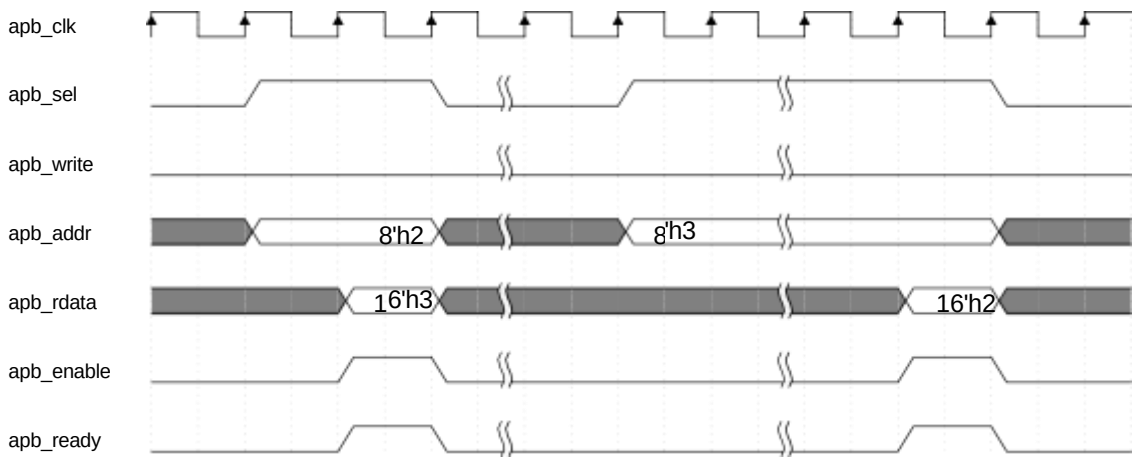


FIGURE 2-23 APB INTERFACE TYPICAL READ TIMING

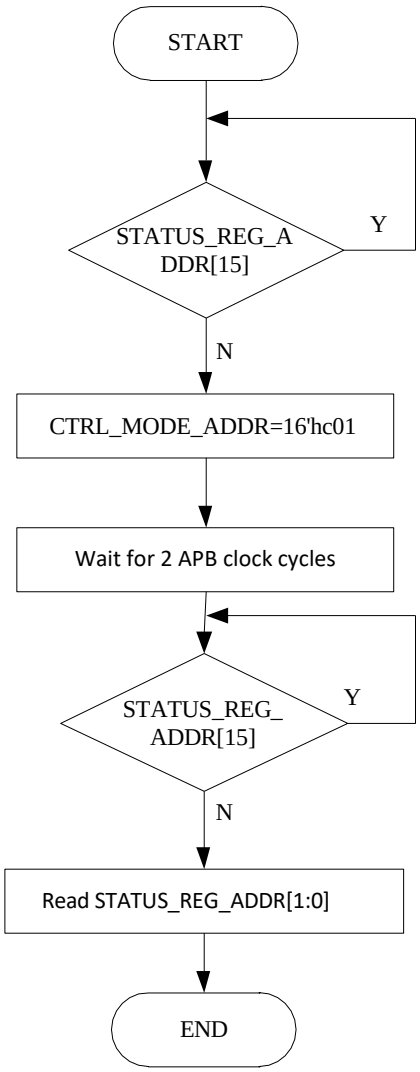
- The first clock cycle: apb_sel is pulled high, apb_write is pulled low, apb_addr is given the initial value, and they need to remain stable until they are released after the handshake is completed.
- Second clock cycle: apb_enable is pulled high until released after the handshake is completed.
- Valid data: apb_rdata is only valid during handshake.

(3) DDR3 Status switching and query methods

In order to prevent the user's state switching request from interfering with the normal DDR3 timing, the user needs to follow certain operating rules when performing state switching (query the STATUS_REG_ADDR register to determine the current state of DDR3 and whether the request has been responded to; control the CTRL_MODE_DATA register to achieve switching between DDR3 states).

- When sending a new status switching request, you must first check whether the current status switching request has been responded to, otherwise an error will occur;
- The state switching request and the state switching trigger enable need to be sent synchronously, that is, bit0 and bit[15:14] of CTRL_MODE_DATA need to be configured at the same time.
- When configuring the Mode Registers inside DDR3, you need to configure the MODE_REG_0_ADDR, MODE_REG_1_ADDR, MODE_REG_2_ADDR, MODE_REG_3_ADDR registers first, and then configure the CTRL_MODE_DATA register.
- After sending the status switching request, you need to wait for 2 apb_clk before checking the relevant status register to see whether the response has been received.

The state request switching process is shown in Figure 2-24, taking controlling DDR3 to enter the Power Down state as an example.



Step1 : Check whether the current status is responded to

Step2 : Send new status request

Step3 : Wait for 2 APBs clock cycle

Step4: Check whether the new status is responded to

Step5: The new request has been responded, read the new status

FIGURE 2-24 DDR3 STATUS REQUEST SWITCHING FLOW CHART

2.5.3 PHY Interface Description

1. Clock and reset interface

Table 2-16 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 level	PLL Lock indication, high level indicates locked.
rst_gpll_lock	O	1	high level	Reset clock GPLL lock indication, high level indicates locked.
ddrphy_sysclk	O	1	-	The system clock of the PHY is output to the working clock of the controller.

Note: " - " means there is no such parameter.

2. DFI interface

Table 2-17 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 level	The read data of the dfi interface is valid.
dfi_rddata	O	DQ_WIDTH*8	-	dfi interface data.
dfi_wrdata_en	I	4	high level	DFI write enable signal.
dfi_wrdata	I	DQ_WIDTH*8	-	DFI Write data.
dfi_wrdata_mask	I	DQ_WIDTH*8/8	high level	DFI Write data byte mask.
dfi_init_complete	O	1	high level	PHY has completed training related operations and is in normal state.
dfi_error	O	1	high level	PHY training Error indication.
dfi_phyupd_req	O	1	-	PHY Request an update .
dfi_phyupd_ack	I	1	-	The feedback signal of dfi_phy_req allows the PHY to update.

Note: " - " means there is no such parameter.

3. Memory Interface

Description:

The Memory interface in this IP is subject to the protocol. If the particle selected by the user contains an interface that is not included in the protocol, please refer to the corresponding particle Datasheet to add it yourself and handle it properly.

Table 2-18 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 Enter the system clock.
mem_cke	O	1	high level	DDR The input system clock is valid.
mem_dm	O	DM_WIDTH	high level	DDR Input data Mask .
mem_odt	O	1	-	DDR ODT .
mem_cs_n	O	1	low level	DDR 's film selections.
mem_ras_n	O	1	low level	Row address enabled.
mem_cas_n	O	1	low level	Column address enabled.
mem_we_n	O	1	low level	DDR Write enable signal.
mem_reset_n	O	1	low level	DDR reset.
mem_dq	I/O	DQ_WIDTH	-	DDR The data.
mem_dqs	I/O	DQS_WIDTH	-	DDR data path clock.
mem_dqs_n	I/O	DQS_WIDTH	-	DDR data path clock.

Note: " - " means there is no such parameter.

4. Debug interface

Table 2-19 Debug interface

Port	I/O	Bit width	Description
Manually debug input signals			
dbg_gate_start	I	1	Reset sequence control, valid on rising edge.
dbg_cpd_start	I	1	Reset sequence control, valid on rising edge.
dbg_ddrphy_rst_n	I	1	Reset sequence control, active low.
dbg_gpll_scan_rst	I	1	Reset sequence control, active high level.
force_samp_position	I	1	The sampling positions of dqsi and dqsin are fixedly enabled and are active at high level. 0: dqsi and dqsin sampling positions change during the training process. 1: The sampling positions of dqsi and dqsin remain unchanged and are always the initial position.
samp_position_dyn_adj	I	1	After the initialization of dqsi and dqsin sampling positions is completed, dynamic adjustment is enabled and the rising edge is valid. If a rising edge is detected, the offset will be superimposed on the basis of the current sampling position. The superimposed offset

			is determined by init_samp_position_even and init_samp_position_odd.
init_samp_position_even	I	8*DQS_WIDT H	dqsi is the offset step of the initial sampling position based on 90°. The highest bit is the sign bit. If the sign bit is 0, it is added, and if the sign bit is 1, it is subtracted. There are DQS_WIDTH groups, each group is 8 bits, and each group can be configured independently. dqsi and dqsin are also valid when the sampling position is dynamically adjusted.
init_samp_position_odd	I	8*DQS_WIDTH	dqsi is the offset step of the initial sampling position based on 90°. The highest bit is the sign bit. If the sign bit is 0, it is added, and if the sign bit is 1, it is subtracted. There are DQS_WIDTH groups, each group is 8 bits, and each group can be configured independently. dqsi and dqsin are also valid when the sampling position is dynamically adjusted.
wrlvl_en	I	1	Write leveling is enabled, high level is active. 0: Skip the Write leveling process and use the value of init_wrlvl_step directly. 1: Write leveling enabled.
init_wrlvl_step	I	8*DQS_WIDTH	The initial step of dqs when doing Write leveling. There are DQS_WIDTH groups, each group is 8 bits, and each group can be configured independently.
wrcal_position_dyn_adj	I	1	Dynamic adjustment of write direction dqs and dq phases is enabled, and the rising edge is valid. When a rising edge is detected, the offset is superimposed on the current phase, and the superimposed offset is determined by init_wrcal_position.
init_wrcal_position	I	MEM_DQS_ WIDTH*8	The offset step of the initial position of dqs and dq phases based on 90°. The highest bit is the sign bit. If the sign bit is 0, it is added, and if the sign bit is 1, it is subtracted. There are DQS_WIDTH groups, each group is 8 bits, and each group can be configured independently. It is also valid when the writing direction dqs and dq phase are dynamically adjusted.
force_read_clk_ctrl	I	1	dqs gate position fixed enable, high level active 0: The dqs gate position changes during the training process.

			1: The position of dqs gate remains unchanged and is always the initial value.
init_slip_step	I	4*DQS_WIDTH	dqs gate initial value of coarse adjustment position. There are DQS_WIDTH groups, each group is 4 bits, and each group can be configured independently.
init_read_clk_ctrl	I	3*DQS_WIDTH	The initial value of dqs gate fine-tuning position. There are DQS_WIDTH groups, each group is 3 bits, and each group can be configured independently.
debug_cpd_offset_adj	I	1	gpll phase adjustment is enabled and is valid on the rising edge. If the rising edge is detected internally, the gpll phase is adjusted based on debug_cpd_offset_dir and debug_cpd_offset based on 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	The command and address signal output delay adjustment of the memory interface is enabled and is active at high level. 0: Command and address signal output delay is generated by the training process. 1: The command and address signal output delay remains unchanged and is always the setting value of force_ck_dly_set_bin.
force_ck_dly_set_bin	I	8	The command and address output signal output delay adjustment port of the memory interface. The port configuration value is multiplied by 5ps, which adds an additional delay value to the command and address output signals.
Commonly used debugging output signals			
Debug data			
debug_data	O	69*DQS_WIDTH	Debug data of each group of DDRPHY, 8bit DQ shares one DDRPHY
debug_calib_ctrl	O	34	Debug data of Trainning status
dbg_slice_status	O	17*MEM_DQS_WIDTH	training state.
dbg_slice_state	O	22*MEM_DQS_WIDTH	training state.

other			
dbg_dll_upd_state	O	2	dll update control status.
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 status.

2.5.4 PHY Interface timing description

1. DFI Interface specification

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

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

The DFI interface used in this design is the same as the standard DFI 3.1 Specification [11]. For differences, see Table 2-20 and Table 2-21.

Table 2-20 DFI of this design Interface and standard DFI difference

Interface	Difference
Control Interface	There are no requirements for tctrl_delay and tcmd_lat.
Write Data Interface	Except for the dfi_wrdata_cs_n signal, the rest are compatible with standard DFI.
Read Data Interface	Only the dfi_rddata and dfi_rddata_valid signals are retained.
Update Interface	Use a custom interface.
Status Interface	Only dfi_init_complete is retained.
Training Interface	N/A .
Low Power Control Interface	N/A .
Error Interface	Compatible.

Table 2-21 DFI Interface and standard DFI Detailed difference comparison

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 applies 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 on DDR3 .
	dfi_reset_n	dfi_reset_n	MC	DFI chip reset .
	dfi_we_n	dfi_we_n	MC	DFI write enable signal , only available for 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 of the dfi interface is valid.
	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 .
	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 Training Signals	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. DFI Interface timing

The working clock frequency ratio of MC to PHY is 1:4, that is, each MC instruction corresponds to 4 Phase PHY instructions. Instructions and data can be distributed among the 4 Phases of the PHY according to the requirements. When users send control instructions through the DFI interface, they must strictly follow the requirements of DDR SDRAM on instruction and data latency. DFI instructions can only be received after dfi_init_complete is pulled high, and the PHY aligns the data for read operations.

(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, you need to send the write command first and then send the data. The typical timing sequence is shown in Figure 2-25.

(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, you need to send a read command first, and then receive the data returned by the PHY through DFI. The typical timing sequence is shown in Figure 2-26.

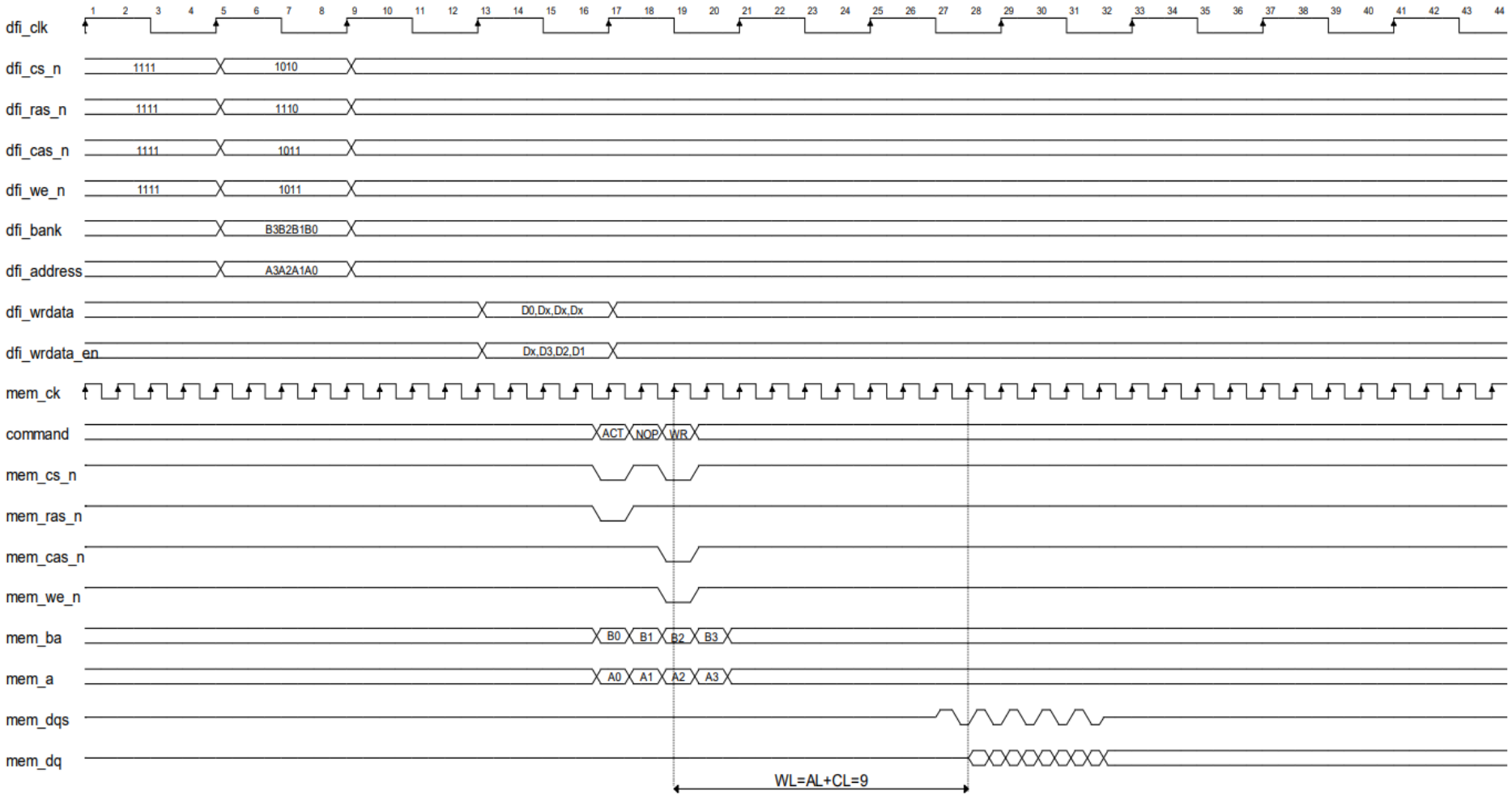


FIGURE 2-25 DFI INTERFACE WRITE OPERATION TIMING

Note: This timing diagram is the simulation timing diagram when WL=9.

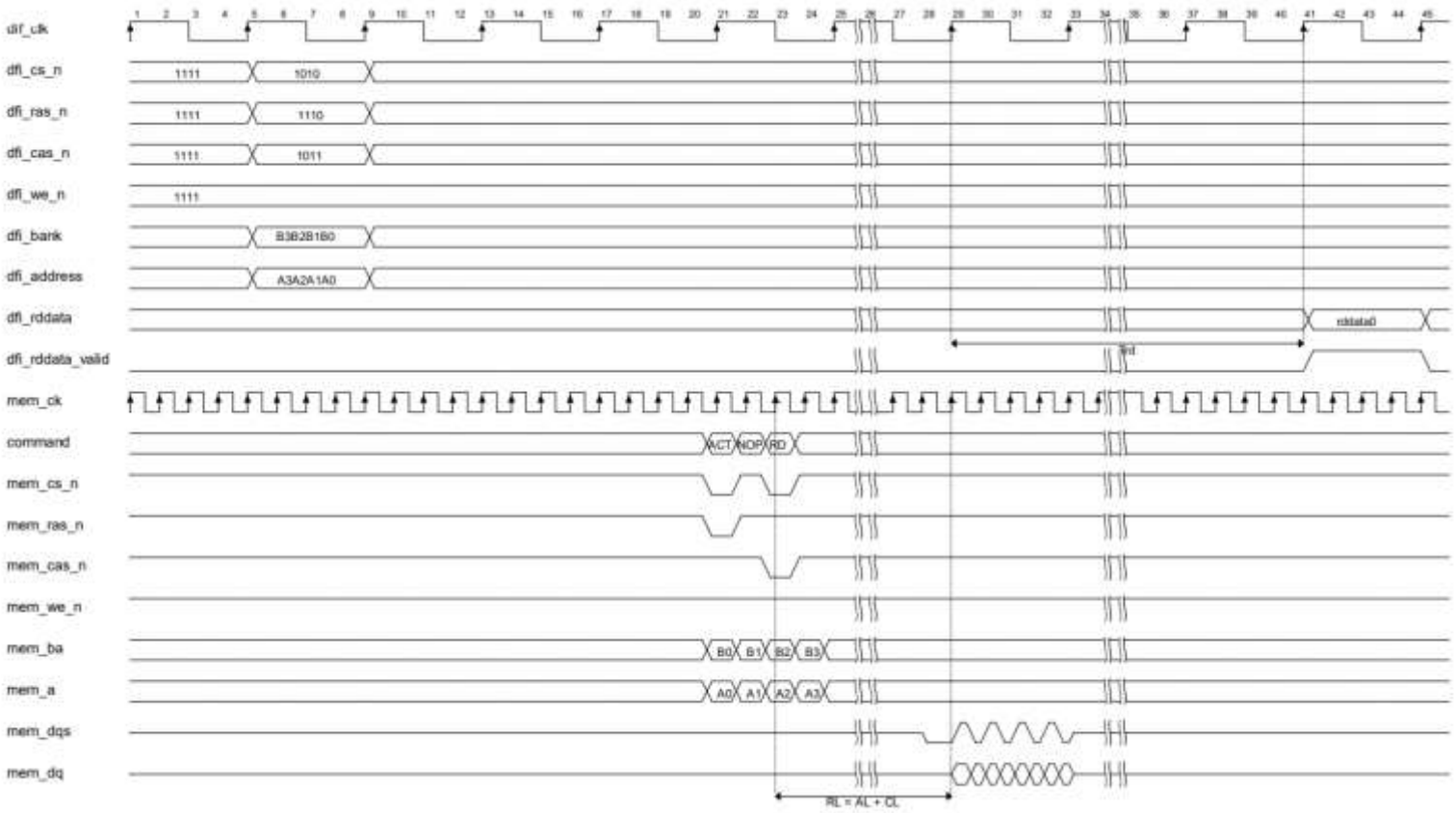


FIGURE 2-26 DFI INTERFACE READ OPERATION TIMING

Note: This timing diagram is a simulation timing diagram, and Trd is 7 dfi_clk cycles. The actual board timing is different from the above diagram.

2.5.5 Debug Data description

1. debug_data description

For a description of the meaning of each debug_data field, see Table 2-22.

Table 2-22 debug_data field definitions

Bits	Corresponding debugging port	Description
3:0	coarse_slip_step	The coarse adjustment position obtained by dqs gate training
6:4	read_clk_ctrl	The fine-tuning position obtained by dqs gate training
10:7	gate_win_size	The number of effective windows for dqs gate training
11	gate_check_pass	dqs gate training completion indication (will be set to 0 after completion)
12	rddata_check_pass	Read calibration data to confirm completion indication (will be set to 0 after completion)
20:13	dqs_even_bin	Read the dqs rising edge delay step value obtained after the eye diagram calibration is completed.
28:21	dqs_odd_bin	Read the dqs falling edge delay step value obtained after the eye diagram calibration is completed.
36:29	total_margin_even	dqs margin of rising edge window
45:38	total_margin_odd	dqs margin of falling edge window
55:48	wrlvl_step	dqs delay step obtained when write leveling is completed
56	Reserved	Reserved
57	wrlvl_dq	The dq value returned when writing 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 beats by which this group's ca signal is delayed
68:67	ck_dqs_diff	The difference in mem_ck beats between this group ca signal and the dqs signal

2. debug_calib_ctrl description

For a description of the meaning of each field in debug_calib_ctrl, see Table 2-23.

Table 2-23 debug_calib_ctrl field definitions

Bits	Corresponding debugging port	Description
0	calib_error	training error indication, caused by write leveling, read calibration error
4:1	dbg_upcal	update control state machine
8:5	dbg_eyecal	Reading eye diagram 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-on initialization process control state machine
29:26	dbg_main	training main state machine

33:30	dbg_error_status	The training status when error occurs. 0: No error; 1: write leveling generates an error; 2: read pattern produces error; 3: gate training produces an error; 4: read training produces error; 5: write training produces an error; 6: eye training produces an error.

3. dbg_slice_status description

The bit width of dbg_slice_status is 4*MEM_DQS_WIDTH+12. When MEM_DQS_WIDTH is 1, the meaning of each corresponding field is descriptiond in Table 2-24.

Table 2-24 dbg_slice_status field definitions

Bits	Corresponding debugging port	Description
0	dll_update_code_done	dll update completion indicator signal. 1: Completed; 0: Not completed;
1	eyecal_move_done	Eye diagram calibration single delay adjustment completion indication signal. 1: Adjustment completed; 0: Adjustment not completed;
2	eyecal_check_pass	Eye diagram calibration complete indication signal. 1 : Eye diagram calibration completed; 0 : Eye diagram calibration not completed;
3	wrcal_move_done	Write calibration single delay adjustment completion indication signal. 1: Adjustment completed; 0: Adjustment not completed;
4	wrcal_check_pass	Write calibration completion indication signal. 1: Write calibration completed; 0: Write calibration not completed;
5	dll_lock_tmp	dll lock indicates that each bit corresponds to a group, and the high bit corresponds to the high group. 1: locked; 0: not locked;
6	dqs_gate_comp_done	dqs gate update completion indicator signal. 1: Completed; 0: Not completed;
7	rddata_check_pass_tmp	After the data comparison passes through dqs gate, each bit corresponds to a group, and the high bit corresponds to the high group. 1: passed; 0: failed;
8	gate_cal_error	dqs gate error indication signal. 1: dqs gate error; 0: dqs gate no error;
9	gate_adj_done	dqs gate single position adjustment completion indication signal. 1: Adjustment completed; 0: Adjustment not completed;
10	gate_check_pass	dqs gate completion indicator signal. 1: dqs gate completed; 0: dqs gate not completed;

11	rdel_move_done	Read the calibration single delay adjustment completion indication signal. 1: Adjustment completed; 0: Adjustment not completed;
12	rdel_calib_error	Read the calibration error indication signal. 1: Calibration error; 0: No error;
13	rdel_calib_done	Read the calibration completion indication signal. 1: Calibration completed; 0: Not completed;
14	adj_rdel_done	Read the calibration initialization adjustment completion indication signal. 1: Adjustment completed; 0: Adjustment not completed;
15	wrlvl_dqs_resp_tmp	Write leveling calibration completion indication signal, each bit corresponds to a group, and the high bit corresponds to the high group. 1: Calibration completed; 0: Not completed;
16	wrlvl_error_tmp	Write leveling calibration error indication signal, each bit corresponds to a group, and the high bit corresponds to the high group. 1: Calibration error; 0: No error;

4. dbg_slice_state description

See Table 2-25 for the meaning description of each field in dbg_slice_state.

Table 2-25 dbg_slice_state field definitions

Bits	Corresponding debugging port	Description
7:0	dbg_slice_rdchk_state	Read calibration data and compare state machine status output.
11:8	dbg_slice_rpbd_ctrl_state	Read calibration control dqs delay state machine status output.
14:12	dbg_slice_gate_state	Adjust the dqs gate delay state machine status output.
18:15	dbg_slice_dq_wrcal_state[3:0]	Control the write direction dq delay state machine status output.
21:19	dbg_slice_wrlvl_state	Controls the write leveling delay state machine status output.

2.6 IP Register description

This section introduces the description and access methods of HMIC_S IP related registers.

2.6.1 Register description

1. Controller internal register

(1) MODE_REG_0_ADDR

The definition of each bit of this register is the same as MR0 in DDR3 SDRAM Standard [8], the bit width is 16 bits, and the access address is 0x00.

Table 2-26 MODE_REG_0_ADDR Bit definition

Bits	Name	Reset value	Access type	Description
1:0	BL	2'b00	R	Burst Length .
2	CL[0]	standard value	R/W	CAS Latency , CL Bit 0 .
3	RBT	standard value	R/W	Read Burst Type .
6:4	CL[3:1]	standard value	R/W	CAS Latency , CL Bit [3: 1] , CL The maximum value is 8 .
7	TM	1'b0	R	Mode .
8	DLL	0	R	DLL Reset .
11:9	WR	standard value	R/W	Write recovery for autoprecharge .
12	PPD	1	R	DLL Control for Precharge P.D.
15:13	reserve	0	R	reserve.

(2) MODE_REG_1_ADDR

The definition of each bit in this register is the same as MR1 in DDR3 SDRAM Standard [8], the bit width is 16 bits, and the access address is 0x01.

Table 2-27 MODE_REG_1_ADDR Bit definition

Bits	Name	Reset value	Access type	Description
0	DLL	1'b0	R	DLL Enable .
1	DIC[0]	1'b0	R/W	Output Driver Impedance Control , DIC No. 0 bit .
2	Rtt_Nom[0]	standard value	R/W	Rtt_Nom , No. 0 bit .
4:3	AL	2'b10	R	Additive Latency .
5	DIC[1]	standard value	R/W	Output Driver Impedance Control , DIC 1st bit .
6	Rtt_Nom[1]	1'b0	R/W	Rtt_Nom 1st bit .
7	Level	1'b0	R	Write leveling enable , refer to JESD79-3E for detailed values .
8	reserve	1'b0	R	reserve.
9	Rtt_Nom[2]	standard value	R/W	Rtt_Nom 2nd bit .
10	reserve	1'b0	R	reserve.

11	TDQS	1'b0	R	TDQS enable .
12	Qoff	1'b0	R	Qoff .
15:13	reserve	3'b000	R	reserve.

(3) MODE_REG_2_ADDR

The definition of each bit of this register is the same as MR2 in DDR3 SDRAM Standard [8], the bit width is 16 bits, and the access address is 0x02.

Table 2-28 MODE_REG_2_ADDR Bit definition

Bits	Name	Reset value	Access type	Description
2:0	PASR	3'b0	R	Partial Array Self-Refresh (Optional) .
5:3	CWL	standard value	R/W	CAS write Latency .
6	ASR	1'b0	R	Auto Self-Refresh .
7	SRT	1'b0	R	Self-Refresh Temperature Rang .
8	reserve	1'b0	R	reserve.
10:9	Rtt_WR	2'b00	R/W	Rtt_WR .
15:11	reserve	5'b0	R	reserve.

(4) MODE_REG_3_ADDR

This register stores each configuration bit of MR3 . The bit width is 16 bits and the access address is 0x03 . For the definition of each bit , see Table 2-29 .

Table 2-29 MODE_REG_3_ADDR Bit definition

Bits	Name	Reset value	Access type	Description
[15:0]	reserve	16'h0	R	For detailed values, refer to JESD79-3E

(5) CTRL_MODE_DATA

This register stores the instructions issued by the user (MRS, Low Power, Normal). The access address is 0x04. For the definition of each bit, see Table 2-30.

Table 2-30 Definition of each bit of CTRL_MODE_DATA

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

(6) STATUS_REG_DATA

This register stores the current status of DDR. The access address is 0x05. For the definition of each bit, see Table 2-31.

Table 2-31 STATUS_REG_DATA Bit definition

Bits	Name	Reset value	Access type	Description
1:0	mode_state	2'b0	R	Stores the current state of DDR3. 2'b00: Normal State; 2'b01:Self-Refresh State; 2'b10:Power Down State; 2'b11: MRS State.
14:2	reserve	13'b0	R	reserve
15	busy	1'b0	R	Whether the current request for ddr3 is responded to: 1'b0: The current request has been responded to 1'b1: The current request has not been processed

2. PHY internal register

DDR PHY internal registers support DDR3, DDR2, and LPDDR, and their meanings are completely consistent with the protocol. For detailed description, please refer to JESD79-3D, DDR3 SDRAM Standard[8]; JESD79-2F, DDR2 SDRAM SPECIFICATION[9]; JESD209B, Low Power Double Data Rate (LPDDR) SDRAM Standard[10].

2.6.2 Register access

The internal registers of the Controller are accessed using the APB interface. The register data is passed through the address through apb_wdata and apb_rdata, and the register address is passed through apb_addr. For the address value of each register, please refer to "2.6.1 Register Description".

The access method of PHY internal registers is the same as the access method of Memory internal Mode Register register.

2.7 Typical application

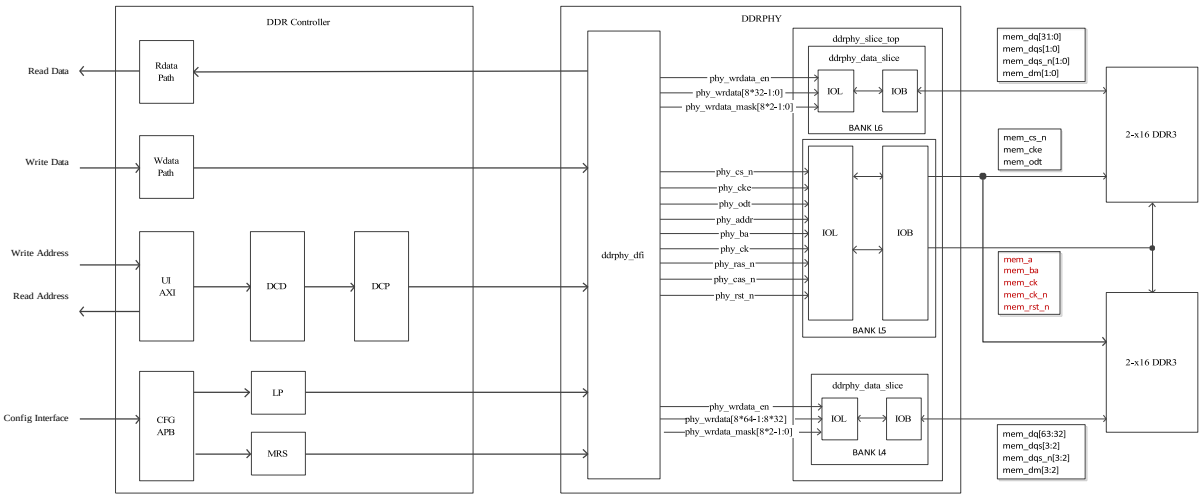


FIGURE 2-27 MULTIPLE BANK-X64 SCHEMATIC

A typical application for multi-BANK-x64 is shown in Figure 2-27. Multi-BANK-x64 is based on two adjacent HR BANKs (L4, L5, L6 in the figure), of which BANK L5 is used for the CA channel, and the other BANK L4 and BANK L6 are used for the DQ channel. Multiple BANK-x64 applications have the following considerations:

- It is prohibited to use the two dedicated pins for calibration resistors in BANK.
- It is recommended that the CA signal is distributed in BANK L5 to ensure that the delay difference of the clock distribution path is short.
- Under the Fly-by topology, try to ensure that the stub wiring of the remote particles meets the design requirements and ensure that the AC parameters can be covered.

2.8 Instructions and Notes

2.8.1 Simplified AXI4 interface Burst calculation

Start_Address = ADDR ;

Burst_Length= len ;

End_Address = ADDR + len * 8 .

2.8.2 Column Address Rounding Example

For example: the Column Address bit width of DDR3 SDRAM is 10, the Length of the user access request is 2, and the Column Address starting address is 10'h3F8, then the address of the second access is 10'h000.

2.8.3 Clock constraints

HMIC_S IP has multiple clocks that need to be constrained, namely ref_clk, rst_clk, ddrphy_sysclk, phy_dq_sysclk (one for each BANK). The relationship between the clocks is shown in Figure 2-28. For specific constraint methods, please refer to the .fdc file in the IP /pnr directory.

Among them, ref_clk is the input reference clock, ddrphy_sysclk and rst_clk are obtained by GPLL frequency multiplication, and phy_dq_sysclk is obtained by PPLL frequency multiplication. ddrphy_sysclk is the system clock of IP soft logic, rst_clk is the driving clock of GPLL dynamic phase shift and reset sequence, phy_dq_sysclk is the BANK clock, each BANK has a PPLL, and the phy_dq_sysclk of the BANK is divided by the PPLL of the BANK.

Notice:

The default format of the ddrphy_sysclk and phy_dq_sysclk timing constraints provided by the .fdc file only supports the ads synthesis tool. If you need to use the OEM synthesis tool, you need to replace it with the OEM format constraints annotated in the .fdc file.

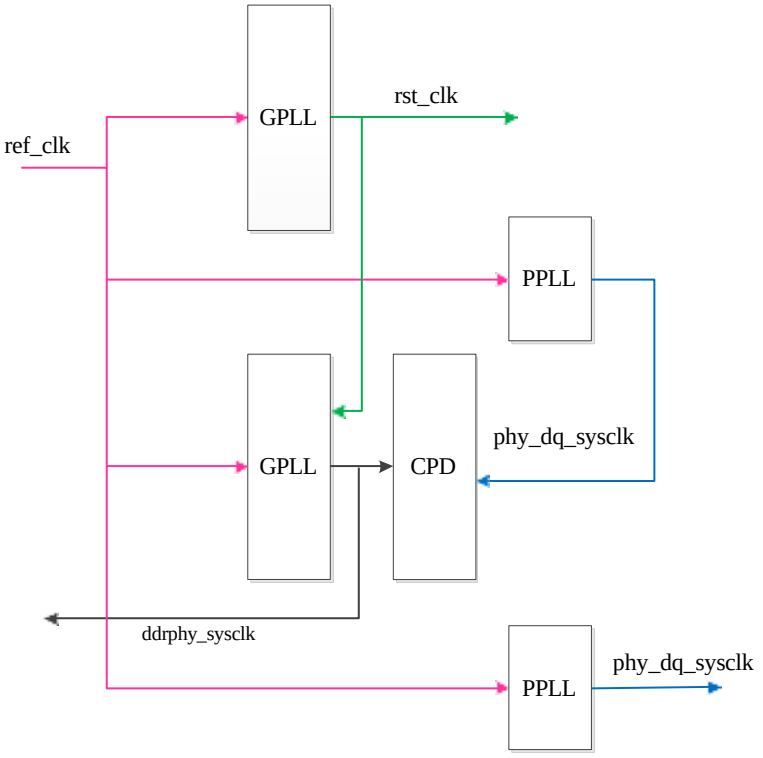


FIGURE 2-28 IP INTERNAL CLOCK STRUCTURE

2.8.4 IO constraint

The position constraints on the internal modules of the chip need to be consistent with the position constraints provided by the .fdc file. The constraint positions cannot be changed, and the paths can be modified according to specific usage conditions.

Constraints on parameters such as pin positions and level standards are performed in UCE (User Constraint Editor). For specific parameter configuration, please refer to the .fdc file in the IP /pnr directory. Some requirements for pin parameter settings are as follows.

- VCCIO needs to be set to 1.5V or 1.8V (1.5 for DDR3, 1.8 for DDR2 and LPDDR);
- mem_ck is a differential output signal and needs to use SSTL15D_I or SSTL18D_I level;
- mem_dqs is a bidirectional differential signal in DDR3 and DDR2 modes and needs to use SSTL15D_I or SSTL18D_I level; it is a single-ended signal in LPDDR mode and needs to use SSTL18_I level;
- The remaining signals are single-ended signals and need to use SSTL15_I and SSTL18 levels;
- mem_dq is a bidirectional single-ended signal, and the reference voltage VREF needs to be configured. The internal VREF is used by default, and VREF_MODE is selected as INT;
- The mem_dqs/mem_dq signal requires terminal resistor matching on the FPGA side. It is recommended to use internal terminal resistors, set DDR_TERM_MODE to ON, and set CP_DYN_TERM to ON;
- Use the default settings for the rest of the parameters.

2.8.5 Routing constraints

DDR projects have requirements for priority routing constraints. Priority routing information is stored in the inst_name.rcf file in the /pnr directory. The Example Design project will automatically load this file. If the user creates the project by himself, the file needs to be loaded manually. Please refer to the help document under the PDS installation path: Route_Constraint_Editor_User_Guide[7].

2.8.6 SCBV set up

When using PG2L100H and PG2T390H products, if any L4 Bank or L5 Bank is unused, SCBV needs to be set at this time. The setting rules are shown in Table 2-32.

Table 2-32 SCBV setting rules

Applicable devices	L4 Bank	L5 Bank	SCBV set up
PG2L100H PG2T390H	Used	Unused	High Voltage (V ≥ 2.5V) Low Voltage （ V ＜ 2.5V ）
	Unused	Used	
	Unused	Unused	High Voltage/Low Voltage

Note: V is the working voltage of the Bank in use.

The SCBV setting interface is shown in Figure 2-29. Select Configuration in the Generate Bitstream column in Project Setting, and you will see the Set Configuration Bank Voltage (SCBV) option, which is divided into Low Voltage and High Voltage. For details, please refer to "UG050012_Titan2 Single Board Hardware Design Guide" [12], "UG040012 Logos2 Single Board Hardware Design User Guide" [13].

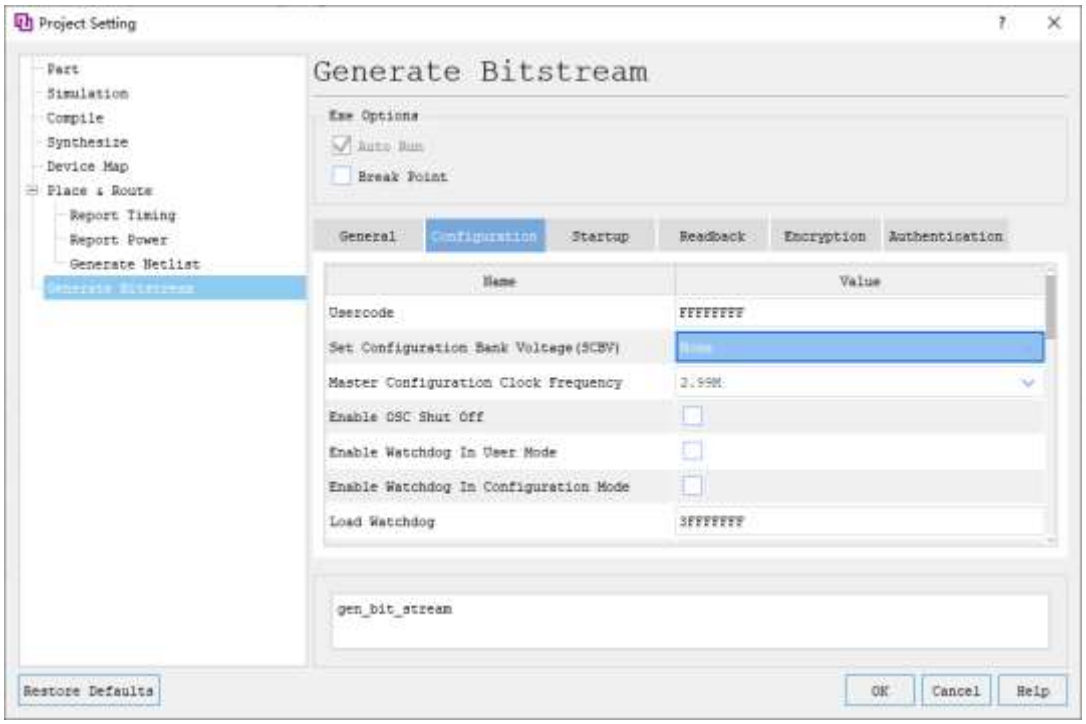


FIGURE 2-29 SCBV SET INTERFACE

2.9 IP Debugging Methods

2.9.1 Key indication signals

For some key information during the operation of DDRPHY, a single-bit indicator signal is made for convenient observation, which can be connected to an external LED or monitored in other ways to quickly determine the operating status of DDRPHY. See Table 2-33 for key indication signal descriptions.

Table 2-33 Key indicators

Port	I/O	Bit width	Description
ddrphy_cpd_lock	O	1	cpd lock indication, high level indicates locked.
ddr_init_done	O	1	PHY has completed training related operations and is in normal state.
heart_beat_led	O	1	Heartbeat signal. When the ddrphy system clock is normal, it flashes once every second. (Signal generated in Example Design)

err_flag_led	O	1	Data detection error signal. Flashing: bist is running normally and there are no errors in data monitoring. 1: There is an error in data detection, which needs to be cleared manually after eliminating the error. 0: bist failed to run properly. (Signal generated in Example Design)

2.9.2 Internal status and control signals

For other non-real-time changing internal status signals and control signals used for debugging, see Table 2-19. These signals can be easily read and written through the serial port. Flexible debugging can be achieved by developing supporting scripts.

For a schematic diagram of serial port access, see "Figure 2-14 Example Design System Block Diagram". The ips2l_uart_ctrl module in DDRPHY Example Design is a serial port communication module. The host computer implements reading and writing operations of internal status signals and control signals through the serial port and ips2l_uart_ctrl module.

The serial port configuration used by DDRPHY Example Design is shown in Table 2-34.

Table 2-34 DDRPHY Example Design Serial port configuration

baud rate	start bit	data bits	Stop bit	Check Digit	flow control
115200bps	1bit	8bit	1bit	None	None

Description:

A set of serial port scripts were developed during the internal debugging process. If necessary, please contact AE to obtain it.

3. Contact us

Microterra Ltd.

fpga@microterra.ru

Chapter 4 Appendix

4.1 Reference documentation

[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 Interface, DFI 3.1 Specification

[12] UG050012_Titan2 Single Board Hardware Design Guide

[13] UG040012 Logos2 Single Board Hardware Design User Guide

4.2 Glossary

A

APB	Advanced Peripheral Bus
AXI	AdvancedeXtensible Interface

D

DDR	Double Data Rate DFI DDR PHY Interface

L

LP	Low Power

M

MC Memory Controller

MR Mode Register

MRS Mode Register Set

P

PHY Physical

R

RD Read

U

UI User Interface

W

WR Write

4.3 abbreviation list

D

DCD DDR Command Decode

DCP DDR3 Command Procedure

H

HMIC High performance Memory Interface Controller

I

IPCIP Compiler

P

PDS Pango Design Suite

PCE Physical Constraint Editor

U

UCE User Constraint Editor

4.4 statement

4.4.1 Copyright Notice

4.4.2 Disclaimer

1. This document only provides phased information, and the content may be updated at any time according to the actual situation of the product without prior notice. Our company does not assume any legal responsibility for direct or indirect losses caused by improper use of this document.

2. This document is provided "as is" without warranty of any kind, including any warranty of merchantability, fitness for a particular purpose, or non-infringement, and any warranty mentioned elsewhere in any proposal, specification or sample. No license, express or implied, to the use of any intellectual property rights is hereby granted by estoppel or otherwise.

3. The company reserves the right to modify documents related to the company's series of products at any time without prior notice.