HMIC_H IP

User Guide

(UG022001, V1.2_i1)

(19.12.2023)

Revisions History

Document Revisions

Version	Date	Revisions	Applicable IP and Corresponding Versions
V1.2_i1	19.12.2023	Initial release.	V1.2

IP Revisions

IP Version	Date	Revisions
V1.2	01.09.2022	Initial release.

About this Manual

Terms and Abbreviations

Terms and Abbreviations	Meaning
APB	Advanced Peripheral Bus
AXI	Advanced eXtensible Interface
DDR	Double Data Rate
DDC	Dedicated DQS Circuit
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
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***

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_H (High performance Memory Interface Controller Hard 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 features and usage of HMIC_H IP.

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.

Chapter 2 IP User Guide

This chapter provides a guide on the use of HMIC_H IP, including an introduction to IP, IP block diagram, IP generation process, Example Design, IP interface description, IP register description, typical applications, descriptions 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_H 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 HMIC_H IP are as follows:

- LPDDR, DDR2, and DDR3 supported;
- x8 x16 Memory Device supported;
- Standard AXI4 bus interfaces:
- (One set of 128bit AXI4 Host Ports and two sets of 64bit AXI4 Host Ports);
- Standard APB bus interface (DDRC configuration interface);
- Configurable low-power mode: Self-Refresh and Power Down;
- Up to 1066Mbps data rate for DDR3;
- Up to 800Mbps data rate for DDR2;
- Up to 400Mbps data rate for LPDDR;
- Burst Length 8 and single Rank.

2.1.2 Applicable Devices and Packages

Table 2-1 HMIC_H IP Applicable Device and Packagings

Applicable Devices	Supported Packages
PGL22G	FBG256/MBG324

2.2 IP Block Diagram

The system block diagram of HMIC_H IP is shown in Figure 2-1.

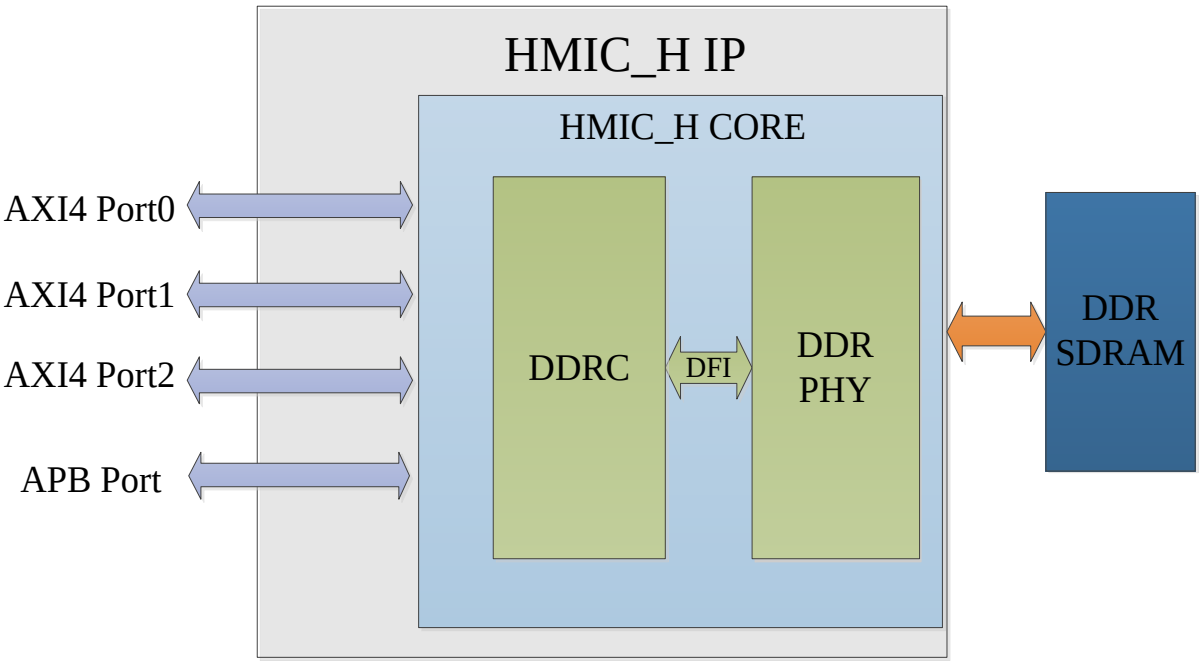

Figure 2-1 HMIC_H IP System Block Diagram

HMIC_H IP includes a DDR Controller, DDR PHY, and PLL. Users can read and write data via the AXI4 interface, configure the internal registers of the DDR Controller via the APB interface, and use PLL to generate various required clocks.

- AXI4 interface

  HMIC_H IP provides three sets of AXI4 Host Ports: AXI4 Port0 (128bit), AXI4 Port1 (64bit), and AXI4 Port2 (64bit). Users can enable these three sets of AXI4 Ports on the HMIC_H IP interface. All three sets of AXI4 Host Ports are standard AXI4 interfaces.

- APB interface

  HMIC_H IP provides an APB configuration interface, through which the internal registers of the DDR Controller can be configured. This interface is selected once the HMIC_H IP initialization is complete.

2.3 IP Generation Process

2.3.1 Module Instantiation

Users can complete the customised configuration of HMIC_H IP through the IPC tool to instantiate 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_H IP module are described as follows.

2.3.1.1 Selecting IP

After selecting the FPGA device type, the Catalog interface displays the loaded IP models. Select Logos HMIC_H under the "System/DDR/Hard" directory, then set the Pathname and Instance Name on the right page. The project instantiation interface is shown in Figure 2-2.

Recommendation:

It is recommended to use version 2022.1-ads or above for the software.

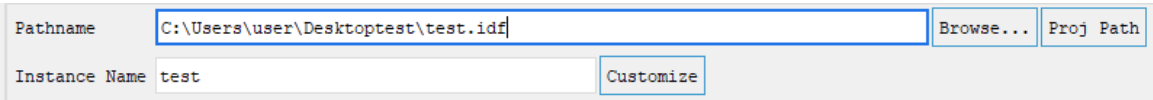

Figure 2-2 Project Instantiation Interface

2.3.1.2 IP Parameter Configuration

After selecting the IP, click <Customize> to enter the HMIC_H IP parameter configuration interface. The left Symbol is the interface block diagram, as shown in Figure 2-3; the Parameter Configuration window is shown on the right side, as shown in Figure 2-4.

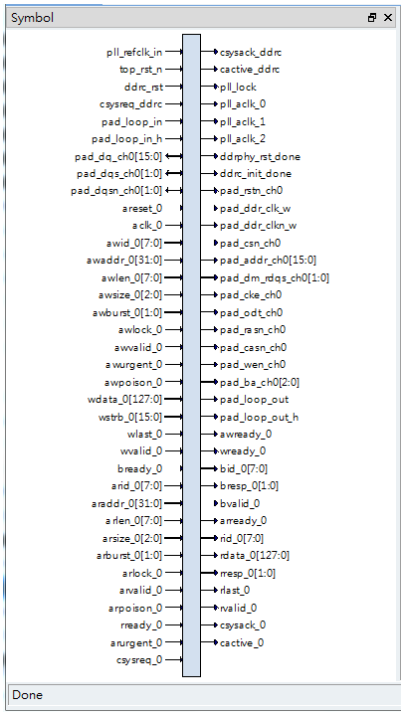

Figure 2-3 HMIC_H IP Interface Block Diagram

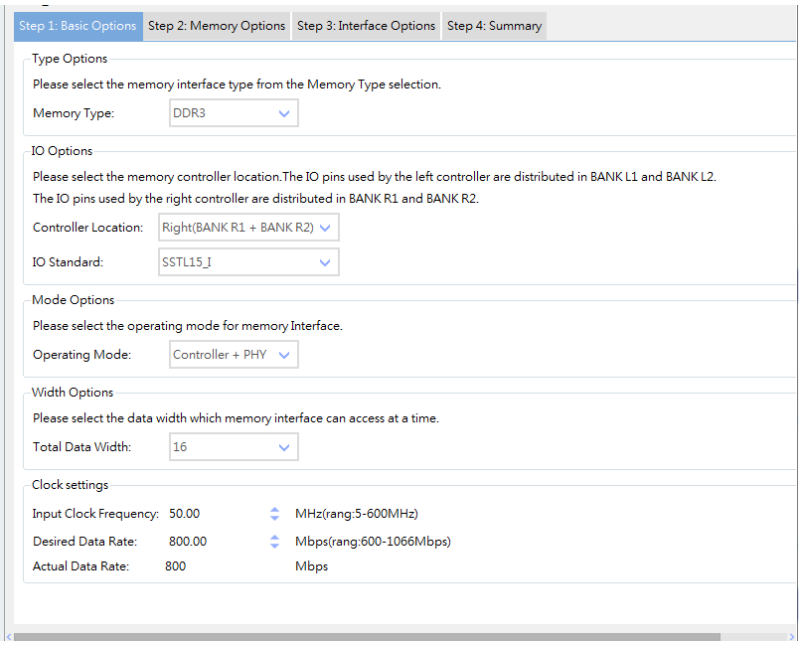

Figure 2-4 HMIC_H IP Parameter Configuration Interface

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, Step2: Memory Options, Step3: Interface Options, Step4: Summary. The steps for configuring HMIC_H 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 2-5. Refer to Table 2-2 for detailed parameters.

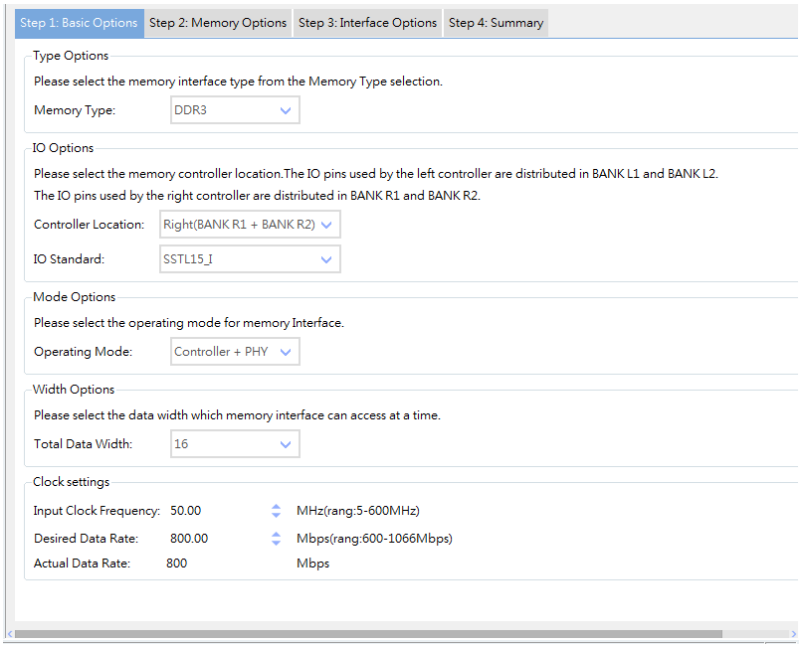

Figure 2-5 Basic Options Page

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

Option Domain	**Parameter/Configuration Options**	**Parameter Description**	**Default Value**
Type Options	Memory Type	The SDRAM type used: Supported types include: 1) DDR3; 2) DDR2; 3) LPDDR;	DDR3
IO Options	Controller Location	The supported locations for the Memory Controller within the FPGA chip include: 1) Right (BANK R1 + BANK R2); 2) Left (BANK L1 + BANK L2); Note: HMIC_H is a hard core, so its location is fixed and cannot be changed.	Right(BANK R1 + BANK R2)

Option Domain	Parameter/Configuration Options	Parameter Description	Default Value
	IO Standard	Interface standard options; The interface standards supported by DDR3 include: 1) SSTL15_I 2) SSTL15_II The interface standards supported by DDR2 and LPDDR include: 1) SSTL18_I 2) SSTL18_II	SSTL15_I
Mode Options	Operating Mode	HMIC_H operating mode selection; Currently, only the Controller + PHY mode is supported;	Controller + PHY
Width Options	Total Data Width	The total DQ width of the off-chip SDRAM connected to HMIC_H. Supported total widths include: 1) 16 2) 8	16
Clock settings	Input Clock Frequency	The input clock of HMIC_H, in MHz.	50.000
	Desired Data Rate	The desired data rate. DDR3, DDR2, and LPDDR support up to 1066Mbps, 800Mbps, and 400Mbps, respectively.	800.000
	Actual Data Rate	The actual data rate, which should be as close to the desired rate as possible.	800.0

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 2-6. Refer to Table 2-3 for detailed parameters.

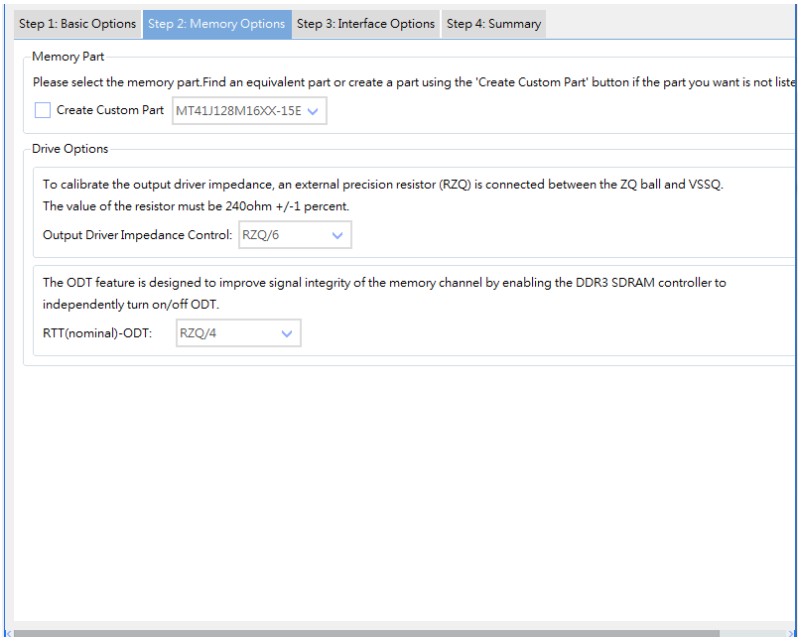

Figure 2-6 Memory Options Page

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

Option Domain	Parameter/ Configuration Options	Parameter Description	Default Value
Memory Part	SDRAM device model	The specific type of SDRAM; The models supported by DDR3 include: 1) MT41J128M8XX-15E 2) MT41J64M16XX-15E 3) MT41J256M8XX-15E 4) MT41J128M16XX-15E 5) MT41J512M8XX-15E 6) MT41J256M16XX-15E The models supported by DDR2 include: 1) MT47H128M8XX-25E 2) MT47H64M16XX-25E 3) MT47H128M16XX-25E 4) MT47H256M4XX-25E 5) MT47H64M8XX-25E-IT The models supported by LPDDR include: 1) MT46H128M16XXXX-5L-IT 2) MT46H64M16XXXX-5L-IT 3) MT46H64M16XXXX-6L-IT 4) MT46H32M16XXXX-5 5) MT46H16M16XXXX-6-IT 6) MT46H16M16XXXX-75-IT Note: If none of these is the required device model, users can check the Create Custom Part option and then customize a new SDRAM type in the Custom Memory Part option box.	MT41J128M16XX-15E
	Create Custom Part	This option box is displayed when the Create Custom Part option is checked, and it is used to customize a new SDRAM type. Options included: Base Part, Timing Parameters, Row Address, Column Address, and Bank Address.	Cleared
Custom Memory Part[1]	Select Base Part	Customizes the reference model of SDRAM;	MT41J128M16XX-15E
	Timing Parameters	Customizes the Timing parameters of SDRAM;	trfc: 160.000 tras: 36.000 trp: 13.500 trcd: 13.500 twr: 15.000 trefi: 7.800 trtp: 7.500 twtr: 7.500
	Row Address	Row address;	14
	Column Address	Column address;	10
	Bank Address	Bank address;	3
Drive Options	Output Driver Impedance Control	Drive strength options; The drive options supported by DDR3 include	RZQ/6

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

Option Domain	Parameter/ Configuration Options	Parameter Description	Default Value
	RTT(nominal)-ODT	(Refer to DDR3 protocol for details) 1) Output Driver Impedance Control 2) RTT(nominal)-ODT The drive options supported by DDR2 include (Refer to DDR2 protocol for details) 1) Output Driver Strength 2) RTT(nominal)-ODT The drive options supported by LPDDR include (Refer to LPDDR protocol for details) 1) Driver Strength	RZQ/4

2.3.1.2.3 Step 3: Interface Options

Interface Options is the configuration page for interface parameters, as shown in Figure 2-7. Refer to Table 2-4 for detailed parameters.

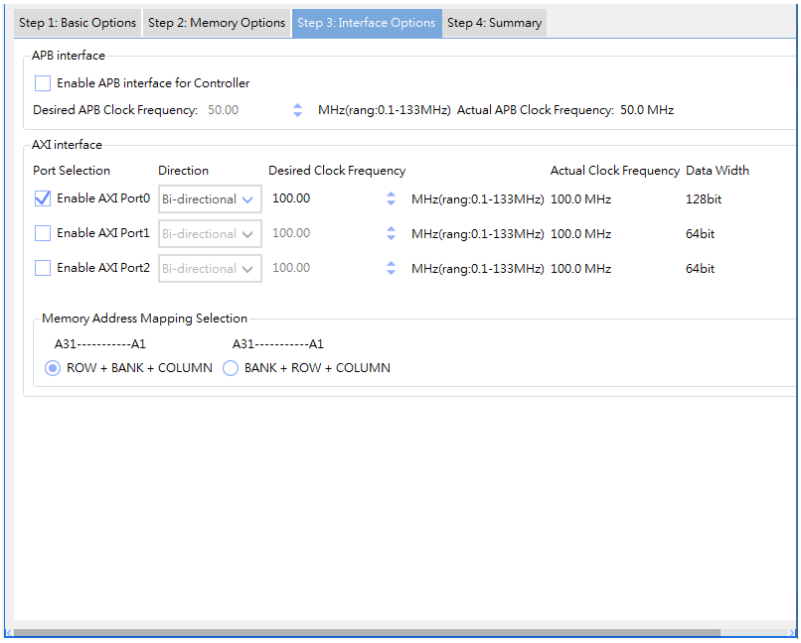

Figure 2-7 Interface Options Page

Table 2-4 Descriptions of Configuration Parameters on the Interface Options Page

Option Domain	Parameter/ Configuration Options	Parameter Description	Default Value
APB Interface	Enable APB Interface for Controller	APB interface configuration, including enable and the clock frequency of the APB interface.	Cleared
AXI Interface	Port Selection	Configuration of three sets of AXI4 interfaces, including enable, read/write direction, and clock frequency. The configuration options for the read/write direction include: 1) Bi-directional read and write 2) Read only 3) Write only	Checked AXI Port0
	Direction		Bi-directional
	Desired Clock Frequency		100.000
Memory Address Mapping Selection	ROW + BANK + COLUMN	Options for mapping SDRAM address to AXI4 address The AXI4 read/write address is 32-bit. However, only bits A1-A31 are valid, and bit A0 is invalid. For example, when ROW+BANK+COLUMN is selected, column address C0 of SDRAM corresponds to AXI address A1, C1 to A2, and so on for other addresses.	Checked
	BANK + ROW + COLUMN		Cleared

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

2.3.1.2.4 Step 4: Summary

The Summary page is used to print the current configuration information without configuration parameters required, as shown in Figure 2-8.

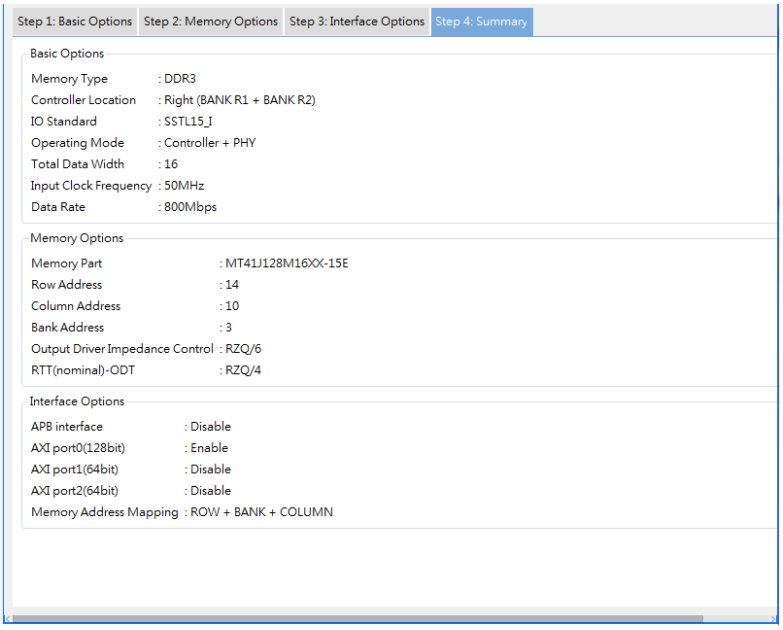

Figure 2-8 Summary Page

2.3.1.3 Generating IP

Upon completion of parameter configuration, click the <Generate> button in the top left corner to generate the HMIC_H IP code according to the user's specific settings. The information report interface for IP generation is shown in Figure 2-9.

Figure 2-9 HMIC_H IP Generation Report Interface

Attention:

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

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

Table 2-5 Output files generated by the IP

Output File[2]	**Description**
$instname.v	The generated IP's top-level .v file.
$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 generated IP's Example Design.
/pnr/*	The project files .pds and pin constraint files .fdc for the generated IP's Example Design.
/sim/*	The simulation directory for the generated IP. ctrl_phy_sim.tcl is a ModelSim simulation script, and sim_file_list.f is the simulation file list.
/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 specific configuration methods of constraint files, please refer to the related help documents under the PDS installation path: "***User_Constraint_Editor_User_Guide***", "***Physical_Constraint_Editor_User_Guide***" and "***Route_Constraint_Editor_User_Guide***".

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

2.3.3 Simulation Runs

The simulation of HMIC_H 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 detailed information about the PDS simulation features and third-party simulation tools, please consult the related help documents in the PDS installation path: "***Pango_Design_Suite_User_Guide***", "***Simulation_User_Guide***".

2.3.3.1 ModelSim Simulation

Simulation operation steps: Open cmd.exe, switch the current directory to "/sim" under the IP generation directory in the command line, run vsim, and open the ModelSim simulation software, as shown in Figure 2-10.

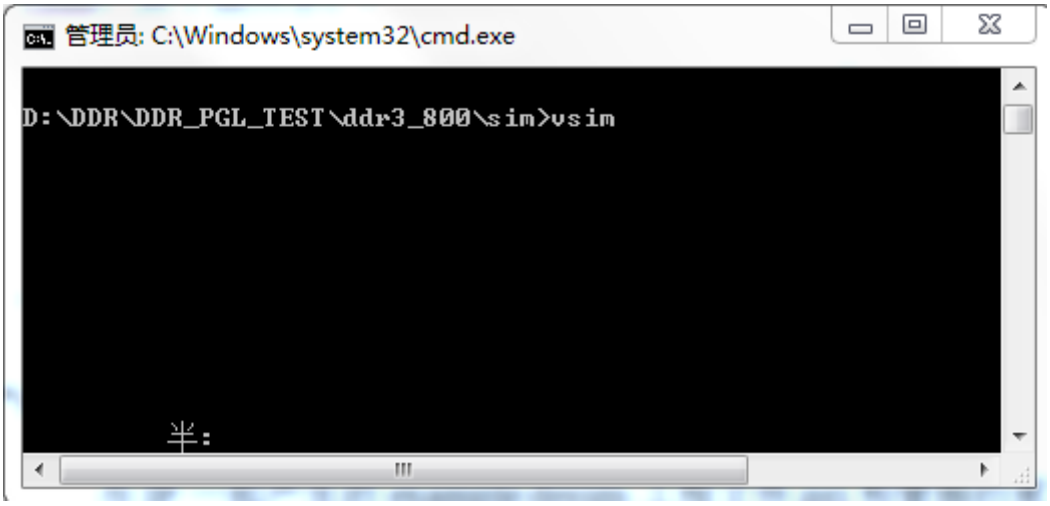

Figure 2-10 Open ModelSim Instruction

Run the sim.tcl script in the ModelSim simulation software to perform simulation, as shown in Figure 2-11.

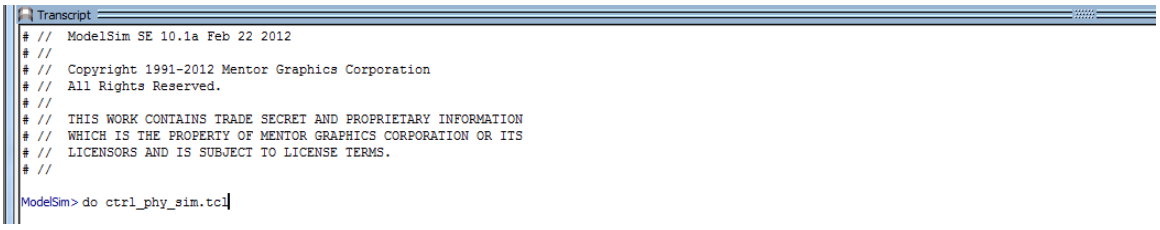

Figure 2-11 ModelSim Simulation Execution Script

2.3.4 Synthesis and Placement/Routing

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

Attention:

The Example Design project files .pds and pin constraint files .fdc generated with the IP are stored in the "/pnr/example_design" directory. Modify the physical constraints according to the devices and PCB traces used, refer to "2.8 Descriptions and Considerations" for details.

2.3.5 Resource Utilization

Table 2-6 Typical Resource Utilisation Values for HMIC_H IP Based on Applicable Devices

Device	Configuration Mode		Typical Resource Utilisation Values							
			Controller				PHY			
			LUT	FF	IO	USCM	LUT	FF	IO	USCM
PGL22G	DDR3 x16	Controller +PHY	215	25	0	0	151	153	54	0

2.4 Example Design

This section primarily introduces the Example Design scheme based on HMIC_H IP. In this scheme, the user logic acts as AXI Master with HMIC_H 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 a data error occurs, the Error LED will light up.

2.4.1 Design Block Diagram

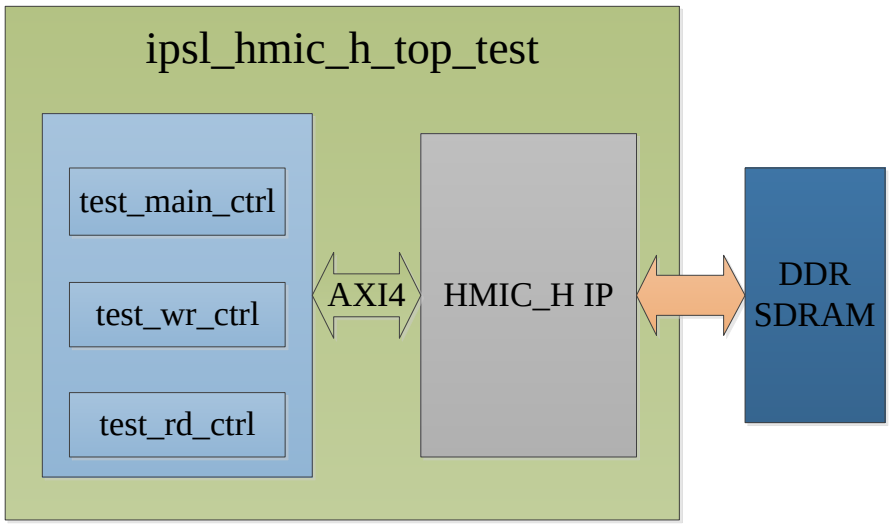

Figure 2-12 Example Design System Block Diagram

As shown in Figure 2-12, the test_main_ctrl module controls AXI read/write instructions, the test_wr_ctrl module controls the AXI instruction writes and data writes, and the test_rd_ctrl module controls the AXI instruction reads and data reads. All three modules support two types of bit widths,

64bit and 128bit. When AXI Port0 (128bit) is used, the 128bit module is called upon; when AXI Port1 (64bit) or AXI Port2 (64bit) is used, the 64bit module is called upon.

2.4.2 Test Method

In the Example Design, the user logic performs read and write operations on HMIC_H IP and validates the readback data. The detailed test process is shown in Figure 2-13.

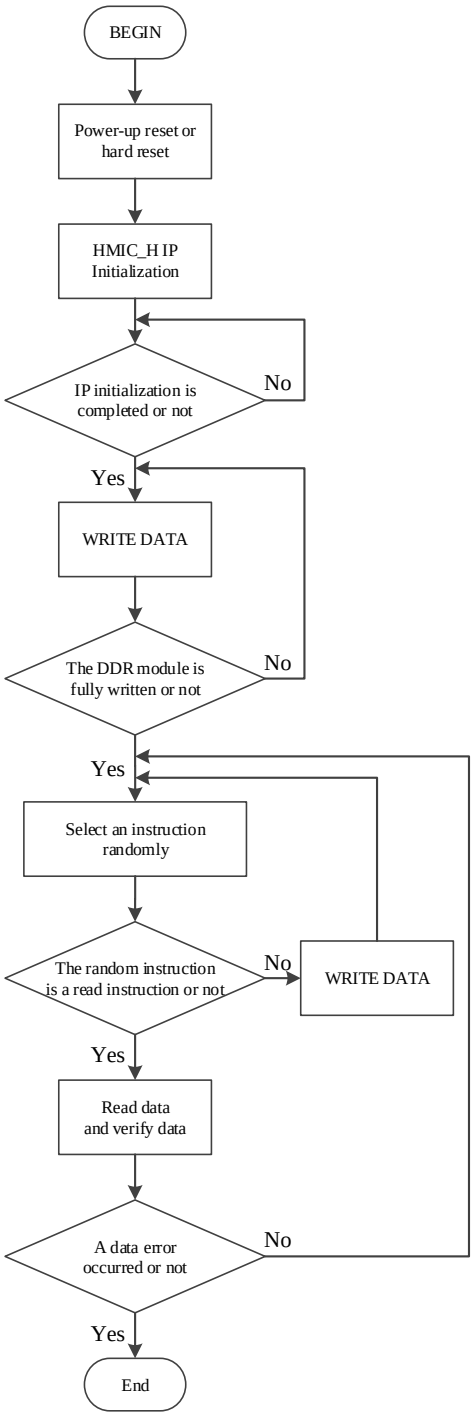

Figure 2-13 Example Design Test Process Diagram

After the system is powered up or a hard reset is initiated, HMIC_H 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 data of the DDR module. After writing full, test_main_ctrl starts random read and write operations, and test_rd_ctrl checks the readback data to determine if any errors occurred.

Attention:

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

2.5 Descriptions of IP Interfaces

This section provides descriptions of interfaces related to HMIC_H IP and timings.

2.5.1 Descriptions of Interfaces

2.5.1.1 Memory Interface

Attention:

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

Table 2-7 Memory Interface

Port Name	I/O	Bit width	Valid Values	Description
pad_addr_ch0	O	15	-	Memory address bus
pad_ba_ch0	O	3	-	Bank address bus
pad_ddr_clk_w	O	1	-	Memory differential clock positive end
pad_ddr_clkn_w	O	1	-	Memory differential clock negative end
pad_cke_ch0	O	1	High	Memory differential clock enable
pad_dm_rdqs_ch0	O	2	High	Data mask
pad_odt_ch0	O	1	-	On Die Termination
pad_csn_ch0	O	1	Low	Memory chip selection
pad_rasn_ch0	O	1	Low	Row address strobe
pad_casn_ch0	O	1	Low	Column address strobe
pad_wen_ch0	O	1	Low	Write enable
pad_rstn_ch0	O	1	Low	Memory reset

Port Name	I/O	Bit width	Valid Values	Description
pad_dq_ch0	I/O	16	-	Data bus
pad_dqs_ch0	I/O	2	-	Data clock positive end
pad_dqsn_ch0	I/O	2	-	Data clock negative end
pad_loop_in	I	1	-	Low temperature compensation input
pad_loop_in_h	I	1	-	High temperature compensation input
pad_loop_out	O	1	-	Low temperature compensation output
pad_loop_out_h	O	1	-	High temperature compensation output

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

2.5.1.2 Global Interface

Table 2-8 Global Interface

Port Name	I/O	Bit width	Valid Values	Description
pll_refclk_in	I	1	-	External reference clock input
top_rst_n	I	1	Low	External reset input
pll_lock	O	1	High	Internal PLL lock signal of HMIC_H
ddrc_rst	I	1	High	Reset input of DDRC
ddrphy_rst_done	O	1	High	Reset completion flag for DDRPHY
ddrc_init_done	O	1	High	Initialization completion flag for DDRC
pll_aclk_0	O	1	-	Axi4 Port0's clock
pll_aclk_1	O	1	-	Axi4 Port1's clock
pll_aclk_2	O	1	-	Axi4 Port2's clock
pll_pclk	O	1	-	APB Port's clock
csysreq_ddrc	I	1	Low	DDRC low power request input
csysack_ddrc	O	1	Low	DDRC low power response
cactive_ddrc	O	1	High	DDRC activation flag

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

2.5.1.3 AXI4 Interface

HMIC_H IP can provide three sets of AXI4 Host Ports: one set of 128-bit ports and two sets of 64-bit ports.

2.5.1.3.1 AXI4 Port0

Table 2-9 AXI4 Port0

Port Name	I/O	Bit width	Valid Values	Description
areset_0	I	1	High	AXI Port0 reset
aclk_0	I	1	-	AXI Port0 input clock

Port Name	I/O	Bit width	Valid Values	Description
awid_0	I	8	-	AXI Port0 Write address ID
awaddr_0	I	32	-	AXI Port0 Write address
awlen_0	I	8	-	AXI Port0 Write Burst length
awsize_0	I	3	-	AXI Port0 Write Burst size
awburst_0	I	2	-	AXI Port0 Write Burst type
awlock_0	I	1	-	AXI Port0 Write Lock type
awvalid_0	I	1	High	AXI Port0 Write address valid
awready_0	O	1	High	AXI Port0 Write address ready
awurgent_0	I	1	High	AXI Port0 Write Urgent. When selected, the port's Write address instruction will be executed preferentially.
awpoison_0	I	1	High	AXI Port0 Write Poison. When selected, the port's Write address instruction will be invalid.
wdata_0	I	128	-	AXI Port0 Write data
wstrb_0	I	16	-	AXI Port0 Write strobes
wlast_0	I	1	-	AXI Port0 Write last
wvalid_0	I	1	High	AXI Port0 Write data valid
wready_0	O	1	High	AXI Port0 Write data ready
bid_0	O	8	-	AXI Port0 Write response ID
bresp_0	O	2	-	AXI Port0 Write response
bvalid_0	O	1	High	AXI Port0 Write response valid
bready_0	I	1	High	AXI Port0 Write response ready
arid_0	I	8	-	AXI Port0 Read address ID
araddr_0	I	32	-	AXI Port0 Read address
arlen_0	I	8	-	AXI Port0 Read Burst length
arsize_0	I	3	-	AXI Port0 Read Burst size
arburst_0	I	2	-	AXI Port0 Read Burst type
arlock_0	I	1	-	AXI Port0 Read Lock type
arvalid_0	I	1	High	AXI Port0 Read address valid
arready_0	O	1	High	AXI Port0 Read address ready
arurgent_0	I	1	High	AXI Port0 Read Urgent. When selected, the port's Read address instruction will be executed preferentially.
arpoison_0	I	1	High	AXI Port0 Read Poison. When selected, the port's Read address instruction will be invalid.
rid_0	O	8	-	AXI Port0 Read data ID
rdata_0	O	128	-	AXI Port0 Read data
rresp_0	O	2	-	AXI Port0 Read response
rlast_0	O	1	-	AXI Port0 Read last
rvalid_0	O	1	High	AXI Port0 Read data valid
rready_0	I	1	High	AXI Port0 Read data ready
csysreq_0	I	1	Low	AXI Port0 enter low power request

Port Name	I/O	Bit width	Valid Values	Description
csysack_0	O	1	Low	AXI Port0 enter low power response
cactive_0	O	1	High	AXI Port0 active

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

2.5.1.3.2 AXI4 Port1

Table 2-10 AXI4 Port1

Port Name	I/O	Bit width	Valid Values	Description
areset_1	I	1	High	AXI Port1 reset
aclk_1	I	1	-	AXI Port1 input clock
awid_1	I	8	-	AXI Port1 Write address ID
awaddr_1	I	32	-	AXI Port1 Write address
awlen_1	I	8	-	AXI Port1 Write Burst length
awsize_1	I	3	-	AXI Port1 Write Burst size
awburst_1	I	2	-	AXI Port1 Write Burst type
awlock_1	I	1	-	AXI Port1 Write Lock type
awvalid_1	I	1	High	AXI Port1 Write address valid
awready_1	O	1	High	AXI Port1 Write address ready
awurgent_1	I	1	High	AXI Port1 Write Urgent. When selected, the port's Write address instruction will be executed preferentially.
awpoison_1	I	1	High	AXI Port1 Write Poison. When selected, the port's Write address instruction will be invalid.
wdata_1	I	64	-	AXI Port1 Write data
wstrb_1	I	8	-	AXI Port1 Write strobes
wlast_1	I	1	-	AXI Port1 Write last
wvalid_1	I	1	High	AXI Port1 Write data valid
wready_1	O	1	High	AXI Port1 Write data ready
bid_1	O	8	-	AXI Port1 Write response ID
bresp_1	O	2	-	AXI Port1 Write response
bvalid_1	O	1	High	AXI Port1 Write response valid
bready_1	I	1	High	AXI Port1 Write response ready
arid_1	I	8	-	AXI Port1 Read address ID
araddr_1	I	32	-	AXI Port1 Read address
arlen_1	I	8	-	AXI Port1 Read Burst length
arsize_1	I	3	-	AXI Port1 Read Burst size
arburst_1	I	2	-	AXI Port1 Read Burst type
arlock_1	I	1	-	AXI Port1 Read Lock type
arvalid_1	I	1	High	AXI Port1 Read address valid
arready_1	O	1	High	AXI Port1 Read address ready

Port Name	I/O	Bit width	Valid Values	Description
arurgent_1	I	1	High	AXI Port1 Read Urgent. When selected, the port's Read address instruction will be executed preferentially.
arpoison_1	I	1	High	AXI Port1 Read Poison. When selected, the port's Read address instruction will be invalid.
rid_1	O	8	-	AXI Port1 Read data ID
rdata_1	O	64	-	AXI Port1 Read data
rresp_1	O	2	-	AXI Port1 Read response
rlast_1	O	1	-	AXI Port1 Read last
rvalid_1	O	1	High	AXI Port1 Read data valid
rready_1	I	1	High	AXI Port1 Read data ready
csysreq_1	I	1	Low	AXI Port1 enter low power request
csysack_1	O	1	Low	AXI Port1 enter low power response
cactive_1	O	1	High	AXI Port1 active

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

2.5.1.3.3 AXI4 Port2

Table 2-11 AXI4 Port2

Port Name	I/O	Bit width	Valid Values	Description
areset_2	I	1	High	AXI Port2 reset
aclk_2	I	1	-	AXI Port2 input clock
awid_2	I	8	-	AXI Port2 Write address ID
awaddr_2	I	32	-	AXI Port2 Write address
awlen_2	I	8	-	AXI Port2 Write Burst length
awsize_2	I	3	-	AXI Port2 Write Burst size
awburst_2	I	2	-	AXI Port2 Write Burst type
awlock_2	I	1	-	AXI Port2 Write Lock type
awvalid_2	I	1	High	AXI Port2 Write address valid
awready_2	O	1	High	AXI Port2 Write address ready
awurgent_2	I	1	High	AXI Port2 Write Urgent. When selected, the port's Write address instruction will be executed preferentially.
awpoison_2	I	1	High	AXI Port2 Write Poison. When selected, the port's Write address instruction will be invalid.
wdata_2	I	64	-	AXI Port2 Write data
wstrb_2	I	8	-	AXI Port2 Write strobes
wlast_2	I	1	-	AXI Port2 Write last
wvalid_2	I	1	High	AXI Port2 Write data valid
wready_2	O	1	High	AXI Port2 Write data ready
bid_2	O	8	-	AXI Port2 Write response ID
bresp_2	O	2	-	AXI Port2 Write response

Port Name	I/O	Bit width	Valid Values	Description
bvalid_2	O	1	High	AXI Port2 Write response valid
bready_2	I	1	High	AXI Port2 Write response ready
arid_2	I	8	-	AXI Port2 Read address ID
araddr_2	I	32	-	AXI Port2 Read address
arlen_2	I	8	-	AXI Port2 Read Burst length
arsize_2	I	3	-	AXI Port2 Read Burst size
arburst_2	I	2	-	AXI Port2 Read Burst type
arlock_2	I	1	-	AXI Port2 Read Lock type
arvalid_2	I	1	High	AXI Port2 Read address valid
arready_2	O	1	High	AXI Port2 Read address ready
arurgent_2	I	1	High	AXI Port2 Read Urgent. When selected, the port's Read address instruction will be executed preferentially.
arpoison_2	I	1	High	AXI Port2 Read Poison. When selected, the port's Read address instruction will be invalid.
rid_2	O	8	-	AXI Port2 Read data ID
rdata_2	O	64	-	AXI Port2 Read data
rresp_2	O	2	-	AXI Port2 Read response
rlast_2	O	1	-	AXI Port2 Read last
rvalid_2	O	1	High	AXI Port2 Read data valid
rready_2	I	1	High	AXI Port2 Read data ready
csysreq_2	I	1	Low	AXI Port2 enter low power request
csysack_2	O	1	Low	AXI Port2 enter low power response
cactive_2	O	1	High	AXI Port2 active

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

2.5.1.4 APB Ports

Table 2-12 APB Interface

Port Name	I/O	Bit width	Valid Values	Description
preset	O	1	High	APB reset
pclk	O	1	-	APB clock
pwdata	I	32	-	APB write data signal
pwrite	I	1	High	APB read/write direction signal
penable	I	1	High	APB enable
psel	I	1	High	APB Peripheral Select.
paddr	I	12	-	APB address
prdata	O	32	-	APB read data signal
pready	O	1	High	APB Ready.

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

2.5.2 Descriptions of Interface Timings

2.5.2.1 AXI4 Interface

The AXI4 interface in this design is a standard AXI4 protocol interface, so users can refer to the AXI4 protocol for its timing.

2.5.2.1.1 AXI4 Single Read Timing

Taking AXI4 Port0 as an example, the timing of a single read operation on the AXI4 interface is shown in Figure 2-14. The master device sends the address, which is then received by the slave device after one cycle. While sending the address, the master device also sends control information indicating the degree and type of Burst. These signals are omitted in the figure for clarity. After the address appears on the address bus, data transfer occurs on the read data channel. The slave device maintains the rvalid_0 signal low until the read data is ready. The device sends the rlast_0 signal indicating the transfer of the last data in this read operation.

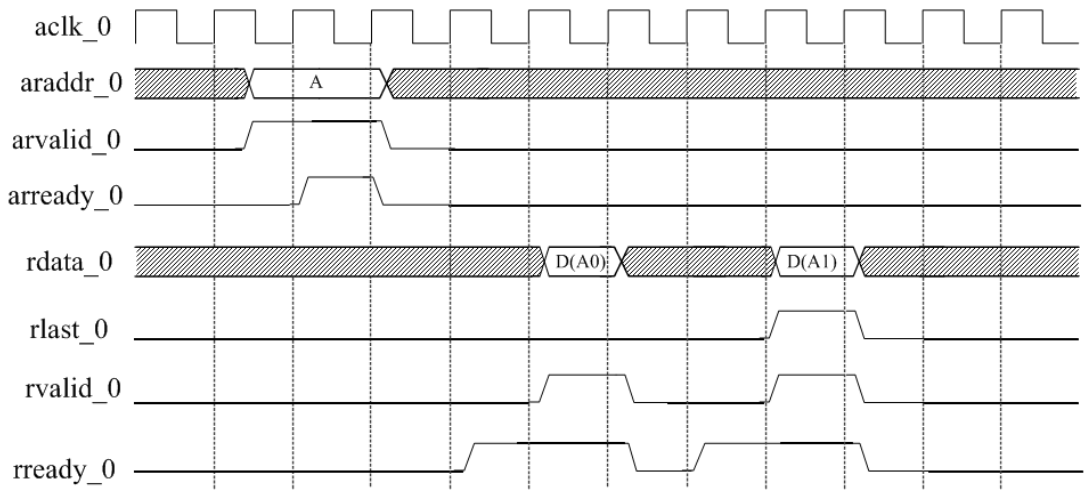

Figure 2-14 AXI4 Single Read Timing

2.5.2.1.2 AXI4 Continuous Read Timing

Taking AXI4 Port0 as an example, the timing of a continuous read operation on the AXI4 interface is shown in Figure 2-15. The master device sends the next read operation's address after receiving the first read operation's address from the slave device. This ensures that a slave device can process the data for the second read operation while completing the first read operation.

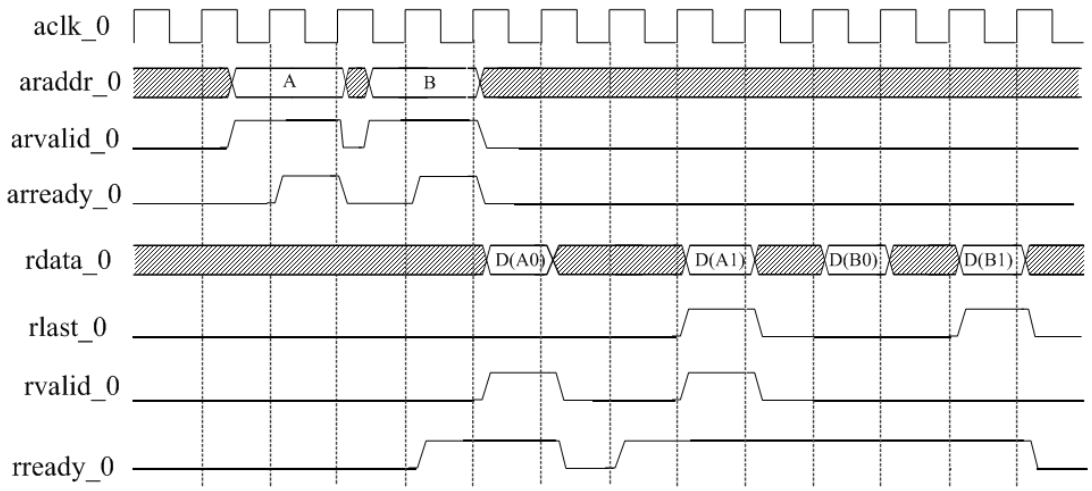

Figure 2-15 AXI4 Continuous Read Timing

2.5.2.1.3 AXI4 Single Write Timing

Taking AXI4 Port0 as an example, the timing of a single write operation on the AXI4 interface is shown in Figure 2-16. The write operation begins after the master device sends the address and control information to the write address channel. The master device then sends each write data through the write data channel, and sets the wlast_0 signal high when sending the last data. When the slave device has received all the data, it will send a write response indicating the end of this write operation to the master device. The continuous write operation is similar to the continuous read operation, i.e., the master device sends the next write operation's address after receiving the first write operation's address from the slave device.

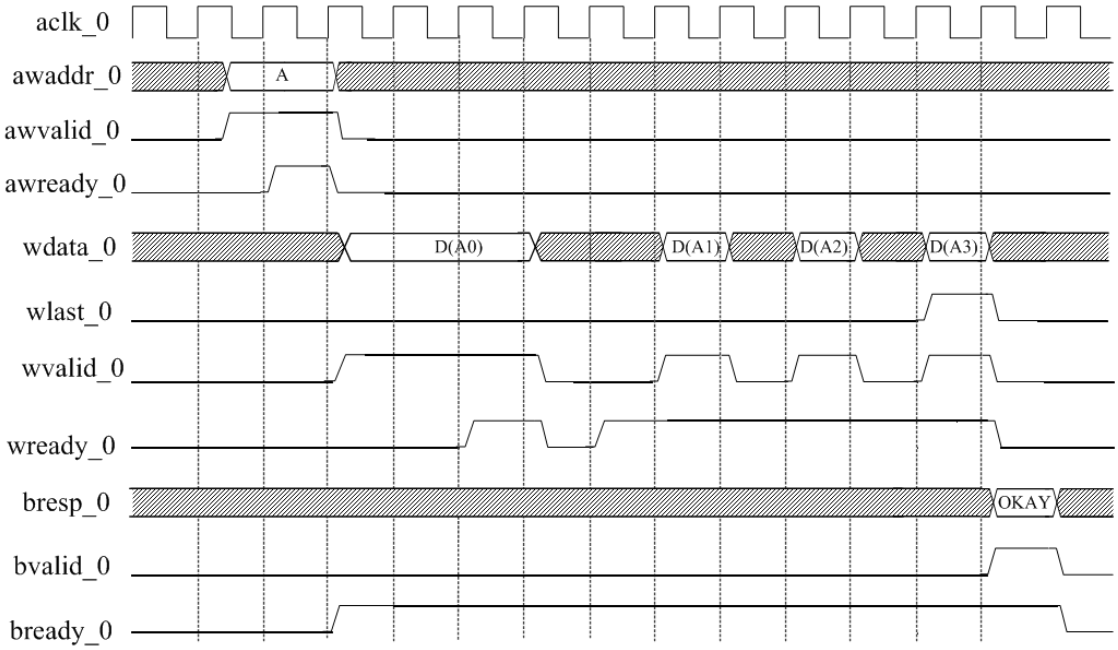

Figure 2-16 AXI4 Single Write Timing

2.5.2.2 APB Ports

The APB interface is for half-duplex operation communication, with independent read and write data lines and multi-funtion control and address lines. Each handshake consumes at least 2 pclk cycles.

2.5.2.2.1 APB interface write timing

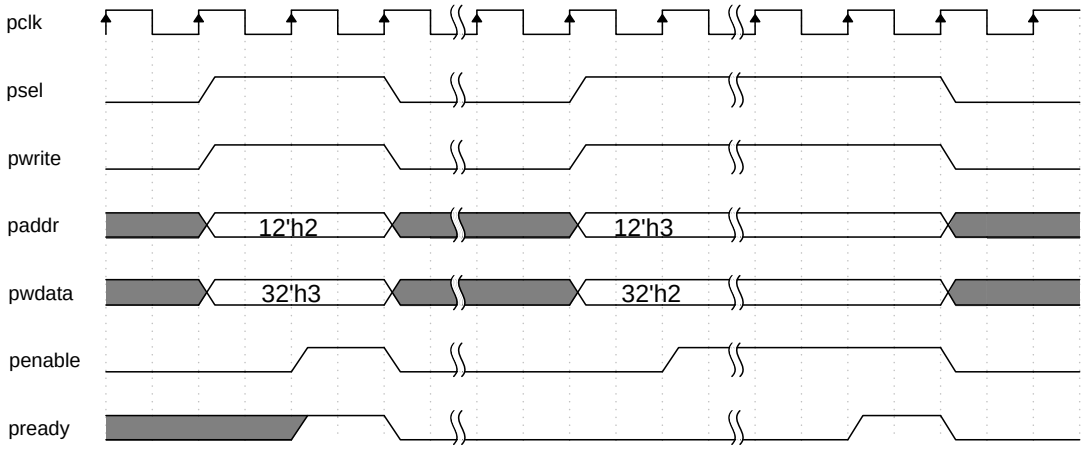

Figure 2-17 Typical Write Timing of the APB Interface

- The first clock cycle: psel and pwrite are pulled high, paddr and pwdata are set to the initial values, which must remain stable until the handshake is completed and then released.

- The second clock cycle: penable is pulled high until the handshake is completed and then released.

2.5.2.2.2 APB interface read timing

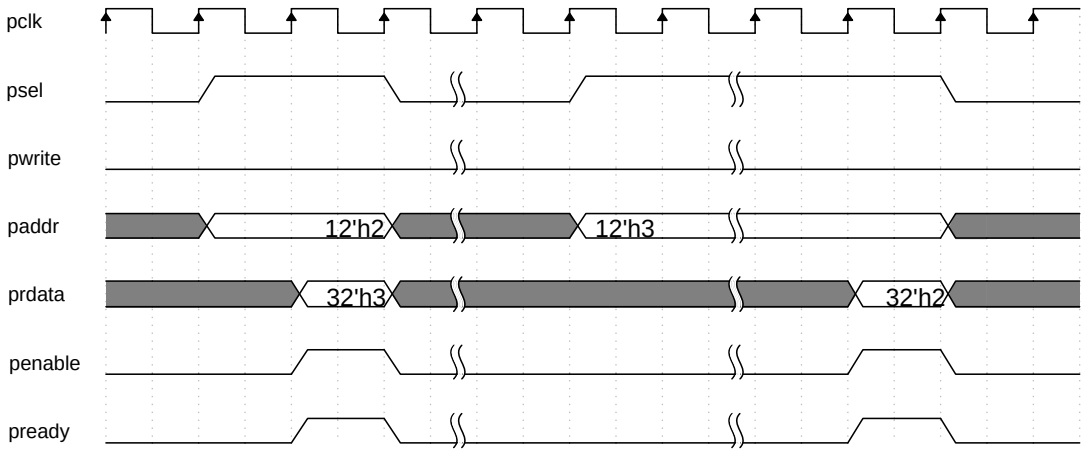

Figure 2-18 Typical Read Timing of the APB Interface

- The first clock cycle: psel is pulled high, pwrite is pulled low, paddr is set to the initial value, which must remain stable until the handshake is completed and then released.
- The second clock cycle: penable is pulled high until the handshake is completed and then released.
- Valid data: prdata is only valid during the handshake.

2.6 Description of the IP Register

HMIC_H IP can read and write the internal configuration registers of DDRC through the APB interface. During the initialization phase, the IP will configure the DDRC's internal configuration registers. Please read and write the DDRC's internal registers after the initialization is completed. Since the IP has correctly configured the internal registers of DDRC during the initialization stage, it is not recommended to arbitrarily change the values of the configuration registers after the initialization is completed.

2.6.1 MSTR

DDRC operating mode configuration register (MSTR) bit width: 16bit, access address: 0x00.

Table 2-13 Definition of Each Bit in MSTR

Bits	Item	Reset Values	Access Type	Description
31:20	-	-	-	Reserved
19:16	burst_rdwr	0x4	R/W	SDRAM burst length used: 0001: Burst length of 2 0010: Burst length of 4 0100: Burst length of 8 1000: Burst length of 16 The IP only supports a Burst length of 8.
15	dll_off_mode	0x0	R/W	DRAM's DLL mode: 0: DLL on 1: DLL off
14	-	-	-	Reserved
13:12	data_bus_width	0x0	R/W	Data bus mode: 00: Full DQ bus width 01: Half DQ bus width 10: Quarter DQ bus width
11:9	-	-	-	Reserved
8	burst_mode	0x0	R/W	burst_mode 0: Sequential burst mode 1: Interleaved burst mode
7:6	-	-	-	Reserved
5	lpddr4	0x0	R/W	Select lpddr4 or not 0: No 1: Yes
4	ddr4	0x0	R/W	Select ddr4 or not 0: No 1: Yes
3	lpddr3	0x0	R/W	Select lpddr3 or not 0: No 1: Yes
2	lpddr2	0x0	R/W	Select lpddr2 or not 0: No 1: Yes
1	mobile ddr	0x0	R/W	Select mobile ddr or not 0: No 1: Yes
0	ddr3	0x1	R/W	Select ddr3 or not 0: No 1: Yes

Note: R/W indicates read/write; R indicates read-only; "-" indicates this parameter does not exist.

2.6.2 STAT

DDRC operating status register (STAT) bit width: 32bit, access address: 0x00.

Table 2-14 Definition of Each Bit in STAT

Bits	Item	Reset Values	Access Type	Description
31:10	-	-	-	Reserved
9:8	selfref_state	0x0	R	Self refresh state. 00: SDRAM is not in Self Refresh 01: Self refresh 1 10: Self refresh power down 11: Self refresh 2
7:6	-	-	-	Reserved
5:4	selfref_type	0x0	R	Indicates whether SDRAM has entered Self Refresh or SR-Powerdown 00: No 11: Yes
3:2	-	-	-	Reserved
1:0	operating_mode	0x0	R	The operating state of DDRC 00: init 01: normal 10: power-down 11: self refresh

Notes: R indicates read-only; "-" indicates this parameter does not exist.

2.7 Typical Applications

For typical applications of HMIC_H IP, please refer to "2.4 Example Design".

2.8 Descriptions and Considerations

2.8.1 Clock Constraints

The IP has 5 clocks, namely pll_refclk_in, phy_clk, pll_aclk0, pll_aclk1, pll_aclk2, and pll_pclk, where pll_refclk_in is the input clock. The clocks phy_clk, pll_aclk0, pll_aclk1, pll_aclk2, and pll_pclk are all derived from the PLL multiplication. The phy_clk is used as the input clock for the HMIC_H hard core, pll_aclk0 as the input clock for AXI4 port0, pll_aclk1 as the input clock for AXI4 port1, pll_aclk2 as the input clock for AXI4 port2, and pll_pclk as the input clock for the APB port. phy_clk is a clock dedicated to HMIC_H. It is used within the IP and cannot be used externally. pll_pclk, pll_aclk_0, pll_aclk_1, and pll_aclk_2 are asynchronous clocks provided for use with external logic, without any phase relationship. The relationship between the clocks is shown in Figure 2-19.

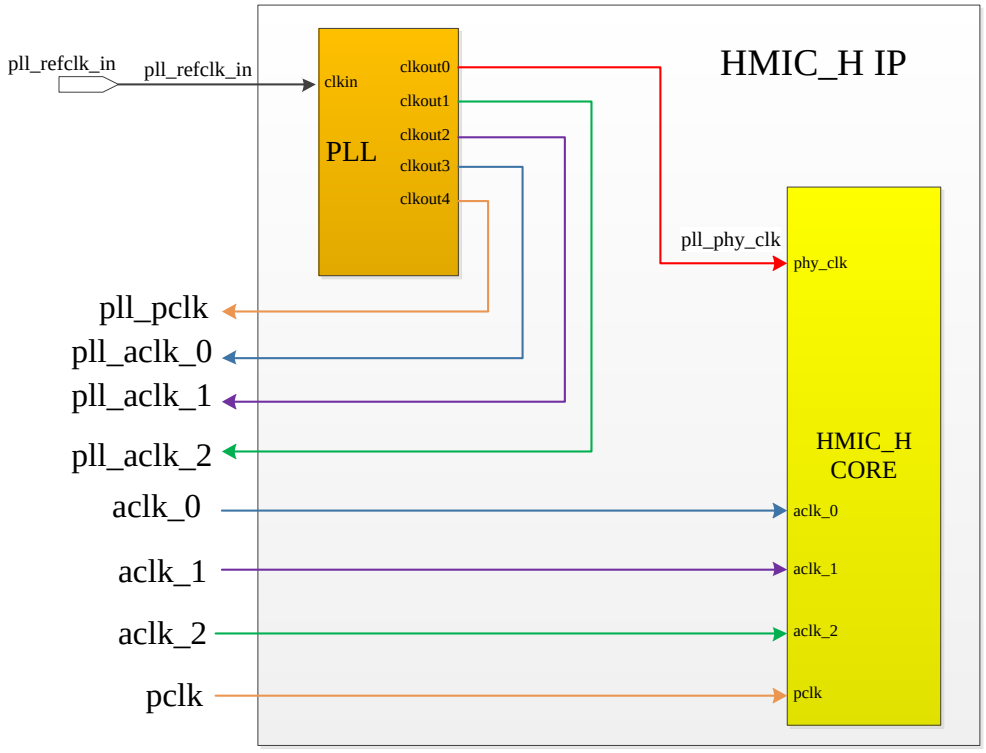

Figure 2-19 IP Clock Relationship Diagram

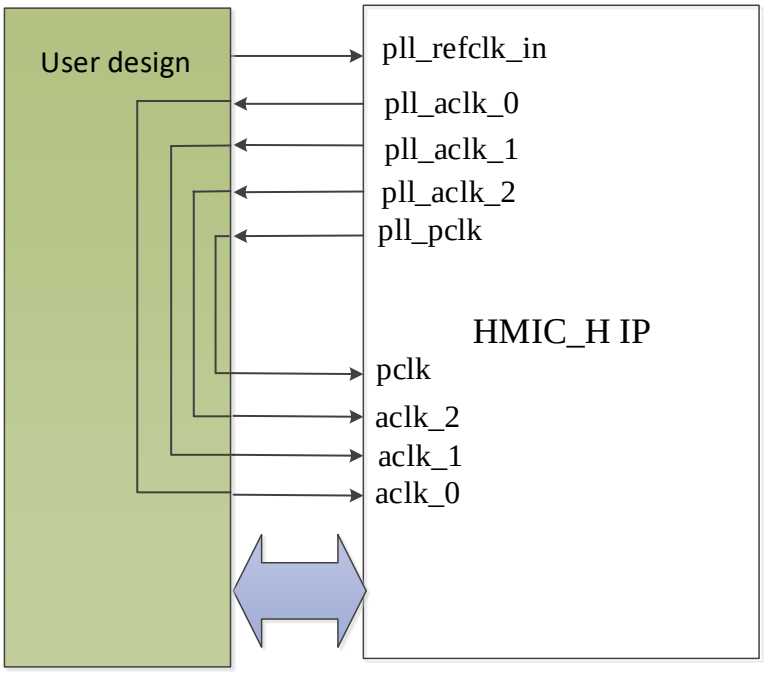

Figure 2-20 IP Clock Connection Diagram

All the 5 clocks within the IP require timing constraints. Take the Example Design as an example.

- Firstly, constrain the external input clock source with the statement as follows:

  create_clock -name {pll_refclk_in} {p:pll_refclk_in} -period {20} -waveform {0 10}

Note: create_clock -name {clock_name} {clock_path} -period {period, in ns} -waveform {time_high}.

- For the PLL's output clock, taking the phy_clk in the Example Design as an example, the constraint statement is as follows:

  create_generated_clock -name {phy_clk} -source {p:pll_refclk_in}
  {t:u_ipsl_hmic_h_top.u_pll_50_400.clkout0} -multiply_by 8

Note: create_generated_clock -name {clock_name} -source {source_of_clock} {clock_path} -multiply_by the multiplication relationship to the source of clock.

- For the specific constraint methods for other clocks, refer to the .fdc file in the "<project_path>/pnr" directory.

2.8.2 Location Constraints

Since HMC_H IP is hard-core, pay attention to the correspondence between the pins and the hard core when using it. The specific precautions are as follows:

- Place the PLL for HMIC_H IP at a specified location. For example, for PGL22GFBG256, the PLL should be fixed at PLL_82_71. Refer to the .fdc file in the "<project_path>/pnr" directory for the constraint method;
- When configuring the corresponding hard core location for HMIC_H IP, if configured as Right, all IOs related to DDR SDRAM must be consistent with the IO constraints in <project_path>/pnr/ddr_xxx_right.fdc (where xxx is the specific package. For example, if packaged as FBG256, then xxx is 256) and cannot be changed; if configured as Left, all IOs related to DDR SDRAM must be consistent with the IO constraints in <project_path>/pnr/ddr_xxx_left.fdc.
- Fix and correspond the I/Os of HMIC_H IP externally connected to DDR SDRAM to DDR3 x16 devices. Therefore, when configuring the Memory Type for HMIC_H IP, if DDR2 and LPDDR are selected, the surplus I/Os of HMIC_H IP should be left floating during PCB production. For instance, if DDR2 does not have an RSTN pin, then leave the pad_rstn_ch0 of HMIC_H IP floating during PCB production.
- The address and data lines of the DDR SDRAM externally connected to HMIC_H IP are fixed-width (x16). If the address or data lines of the used DDR SDRAM are less than those externally connected to HMIC_H IP, the surplus address and data lines of HMIC_H IP should be left floating. For example, when HMIC_H IP is configured with a data width of 8, the higher-bit data lines and clock lines of HMIC_H IP should be left floating.
- pad_loop_in, pad_loop_in_h, pad_loop_out, and pad_loop_out_h are temperature compensation I/Os and should be left floating during PCB production.

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 series products at any time without prior notice.