IGLOO2 HPMS

DDR Controller Configuration

Libero SoC v11.6 and later

Microsemi

Table of Contents

Introduction

The IGLOO2 HPMS has an embedded DDR controller (HPMS DDR). This DDR controller is intended to control an off-chip DDR memory. The HPMS DDR controller can be accessed from the HPMS (using HPDMA) as well as from the FPGA fabric.

When you use System Builder to build a system block which includes an HPMS DDR, System Builder configures the HPMS DDR controller for you based on your entries and selections.

No separate HPMS DDR configuration by the user is required. For details, please refer to the IGLOO2 High Speed DDR Interface User Guide.

System Builder

Invoke System Builder (Figure 1) to configure the HPMS DDR automatically.

Enter the DDR memory setting time. This is the time the DDR memory requires to initialize. The default value is 200 us. Refer to your DDR Memory Data Sheet for the correct value to enter.

1. In the Device Features page of System Builder, check **HPMS External DDR Memory** and then check **MDDR**.

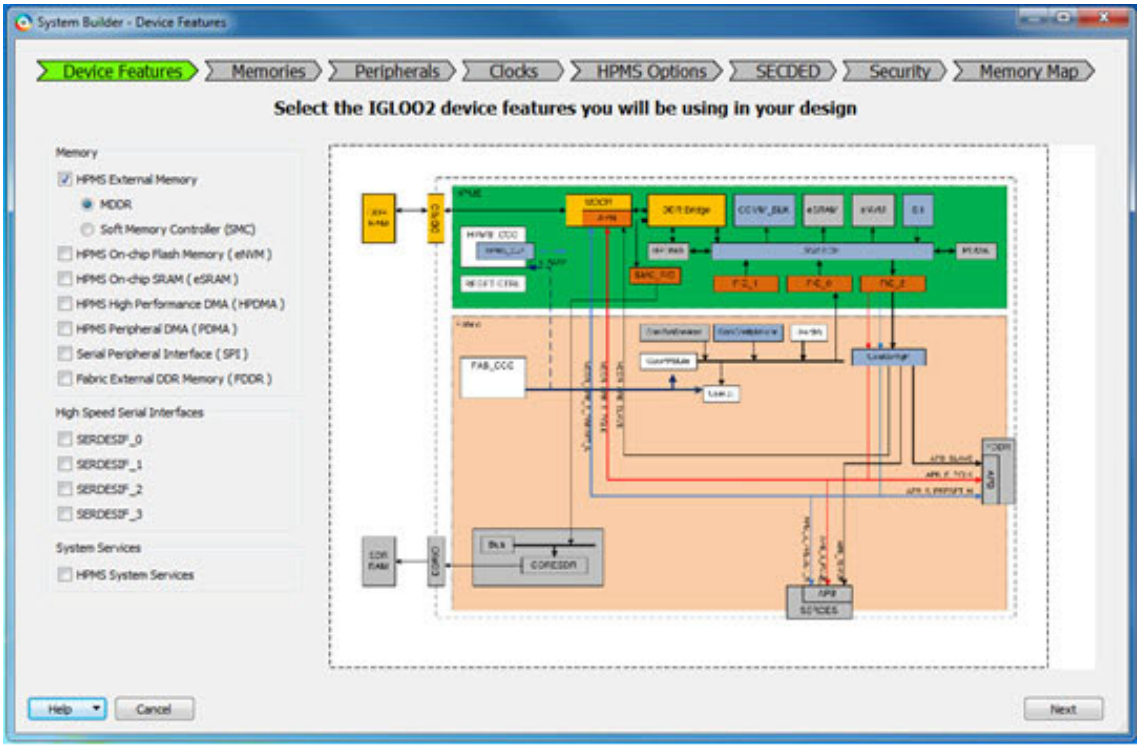

***Figure 1* • System Builder Device Features Page**

2. In the Memories page (Figure 2), select one of the following DDR Memory Types:
   - DDR2
   - DDR3
   - LPDDR
3. Select the **Width** of the DDR Memory: 8, 16 or 32

4. Check **ECC** if you want to have ECC for the DDR.
5. **Arbitration Scheme:** Type-0, Type -1, Type-2, Type-3
6. **Highest Priority ID** - Valid values are from 0 through 15
7. **Address Width (bits)** - Refer to the Data Sheet of your DDR memory for the number of row, bank, and column address bits for the LPDDR/DDR2/DDR3 memory you use. Click the pull-down menu to choose the correct value for rows/banks/columns as per the data sheet of the LPDDR/DDR2/DDR3 memory.

Note: The number in the pull-down list refers to the number of Address bits, not the absolute number of rows/banks/columns. Take for example, if your DDR memory has 4 banks, select 2 (2^2=4) for banks. If your DDR memory has 8 banks, select 3 (2^3=8) for banks.

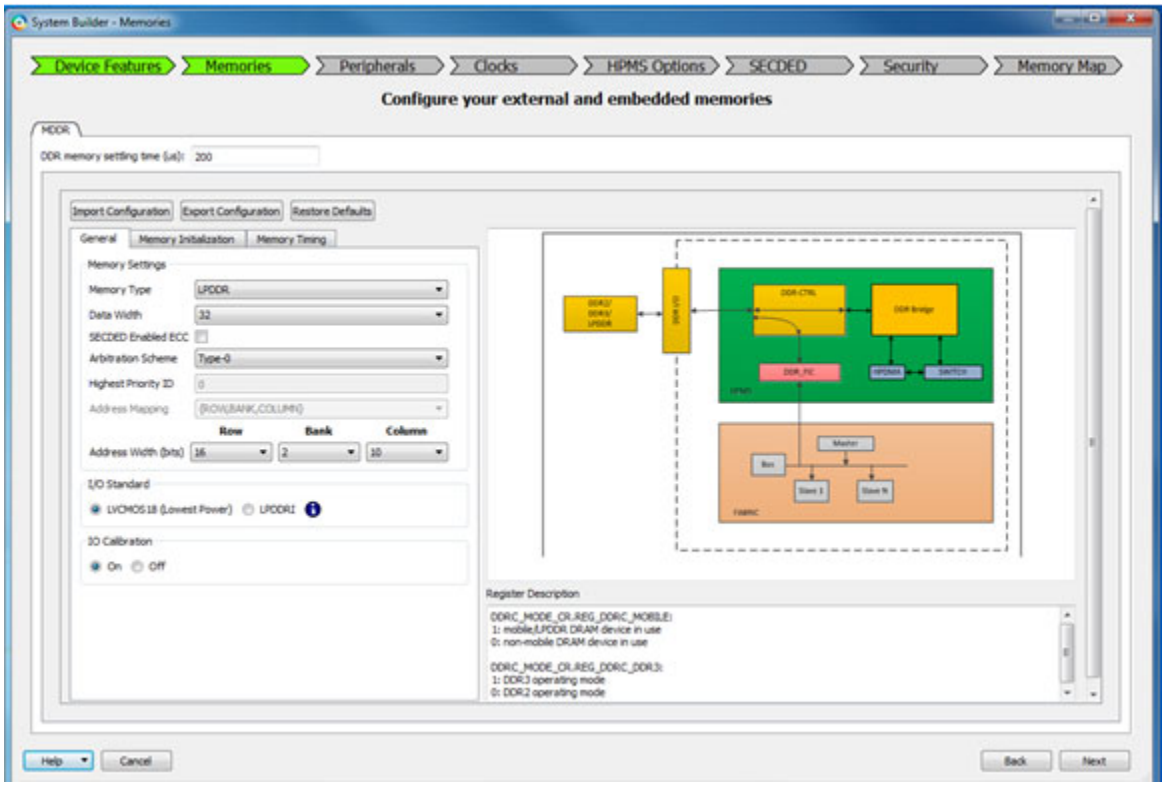

***Figure 2* • System Builder Memories Page**

I/O Drive Strength (DDR2 and DDR3 only)

Select one of the following drive strengths for your DDR I/O's:

- Half Drive Strength
- Full Drive Strength

Libero SoC sets the DDR I/O Standard for your MDDR system based on your DDR Memory type and I/O Drive Strength (as shown in Table 1).

Table 1 • **I/O Drive Strength and DDR Memory Type**

DDR Memory Type	Half Drive Strength	Full Drive Strength
DDR3	SSTL15I	SSTL15II
DDR2	SSTL18I	SSTL18II
LPDDR	LPDRI	LPDRII

IO Standard (LPDDR only)

Select one of the following options:

- LVCMOS18 (Lowest Power) for LVCMOS 1.8V IO standard. Used in typical LPDDR1 applications.
- LPDDRI Note: Before you choose this standard, make sure that your board supports this standard. You must use this option when targeting the M2S-EVAL-KIT or the SF2-STARTER-KIT boards. LPDDRI IO standards require that a IMP_CALIB resistor is installed on the board.

IO Calibration (LPDDR only)

Choose one of the following options when using LVCMOS18 IO standard:

- On
- Off (Typical)

Calibration ON and OFF optionally controls the use of an IO calibration block that calibrates the IO drivers to an external resistor. When OFF, the device uses a preset IO driver adjustment.

When ON, this requires a 150-ohm IMP_CALIB resistor to be installed on the PCB.

This is used to calibrate the IO to the PCB characteristics. However, when set to ON, a resistor needs to be installed or the memory controller will not initialize.

For more information, refer to AC393-SmartFusion2 and IGLOO2 Board Design Guidelines Application Note and the IGLOO2 High Speed DDR Interfaces User Guide.

1 – MDDR Controller Configuration

When you use the HPMS MDDR Controller to access an external DDR Memory, the DDR Controller must be configured at runtime. This is done by writing configuration data to dedicated DDR controller configuration registers. This configuration data is dependent on the characteristics of the external DDR memory and your application. This section describes how to enter these configuration parameters in the MDDR controller configurator and how the configuration data is managed as part of the overall Peripheral Initialization solution.

HPMS MDDR Control Registers

The MDDR Controller has a set of registers that need to be configured at runtime. The configuration values for these registers represent different parameters (for example, DDR mode, PHY width, burst mode, ECC, etc.). For details about the DDR controller configuration registers, refer to the IGLOO2 FPGA High Speed DDR Interfaces User Guide.

MDDR Registers Configuration

Use the Memory Initialization (Figure 1-1, Figure 1-2, and Figure 1-3) and Memory Timing (Figure 1-4) tabs to enter parameters that correspond to your DDR Memory and application. Values you enter in these tabs are automatically translated to the appropriate register values. When you click a specific parameter, its corresponding register is described in the Register Description pane (lower portion in Figure 2 on page 4).

Memory Initialization

The Memory Initialization tab allows you to configure the ways you want your LPDDR/DDR2/DDR3 memories initialized. The menu and options available in the Memory Initialization tab vary with the type of DDR memory (LPDDR/DDR2/DDR3) you use.

Refer to your DDR Memory Data Sheet when you configure the options.

When you change or enter a value, the Register Description pane gives you the register name and register value that is updated. Invalid values are flagged as warnings.

Figure 1-1, Figure 1-2, and Figure 1-3 show the Initialization tab for LPDDR, DDR2 and DDR3, respectively.

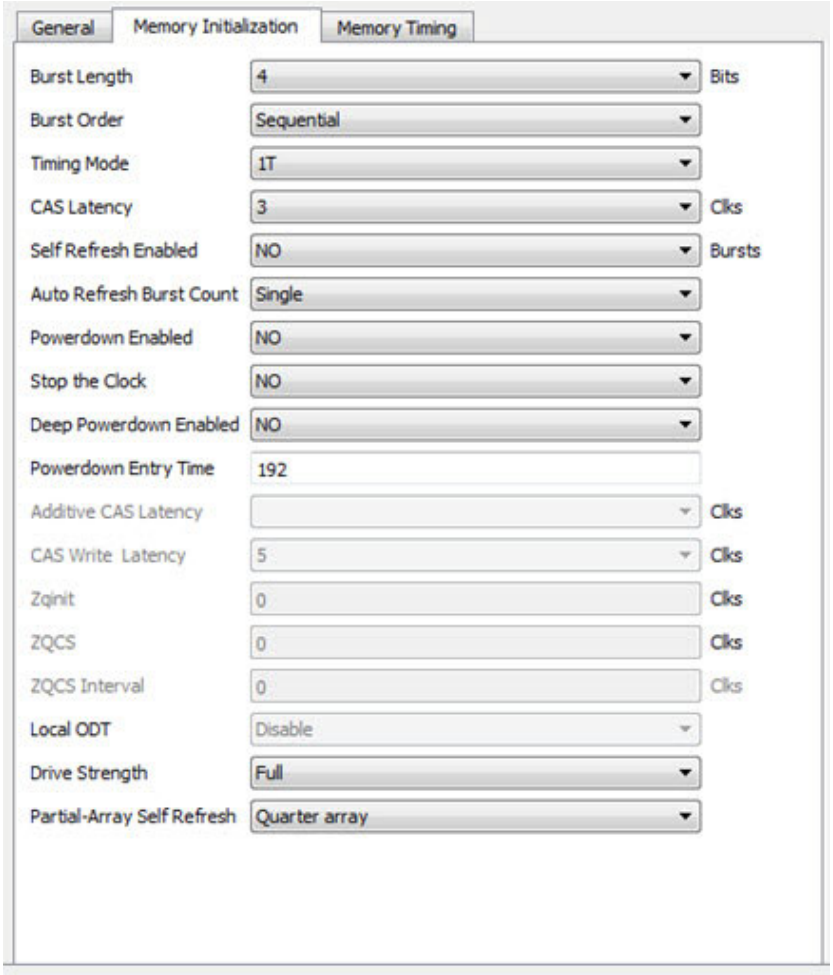

***Figure 1-1* • MDDR Configuration—Memory Initialization Parameters (LPDDR)**

- **Timing Mode** - Select 1T or 2T Timing mode. In 1T (the default mode), the DDR controller can issue a new command on every clock cycle. In 2T timing mode, the DDR controller holds the address and command bus valid for two clock cycles. This reduces the efficiency of the bus to one command per two clocks, but it doubles the amount of setup and hold time.
- **Partial-Array Self Refresh (LPDDR only)**. This feature is for power saving for the LPDDR. Select one of the following for the controller to refresh the amount of memory during a self refresh:
  - Full array: Banks 0, 1,2, and 3
  - Half array: Banks 0 and 1
  - Quarter array: Bank 0
  - One-eighth array: Bank 0 with row address MSB=0
  - One-sixteenth array: Bank 0 with row address MSB and MSB-1 both equal to 0.

For all other options, refer to your DDR Memory Data Sheet when you configure the options.

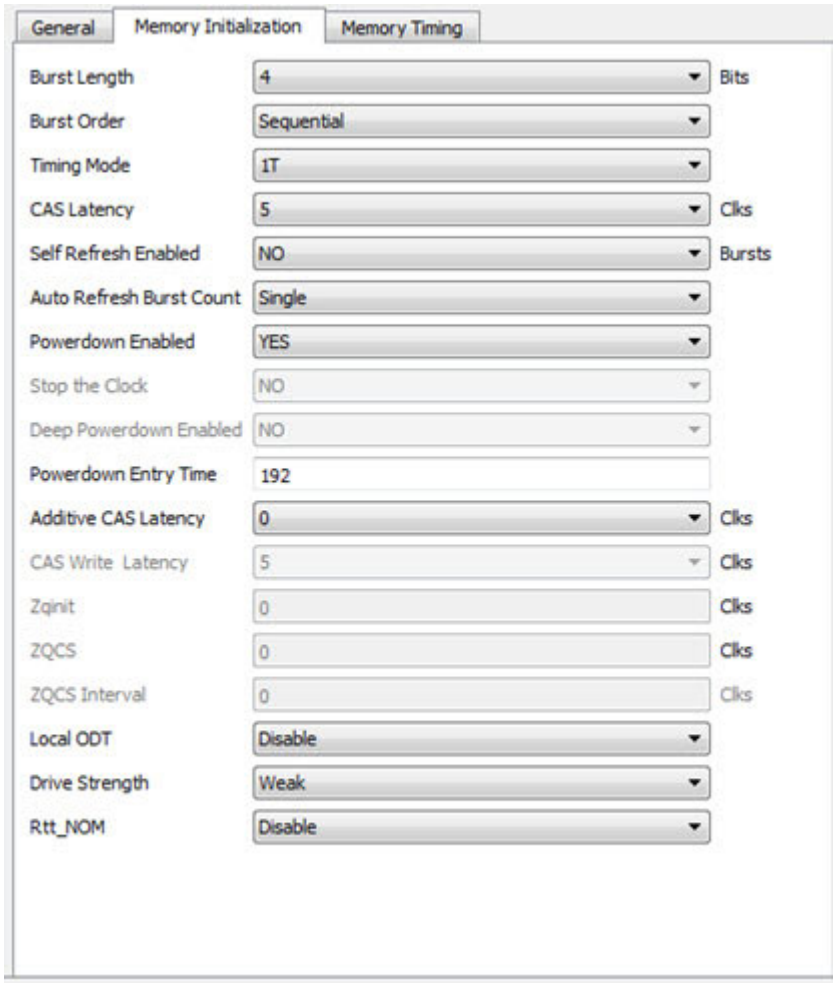

Figure 1-2 **• MDDR Configuration—Memory Initialization Paramet ers (DDR2)**

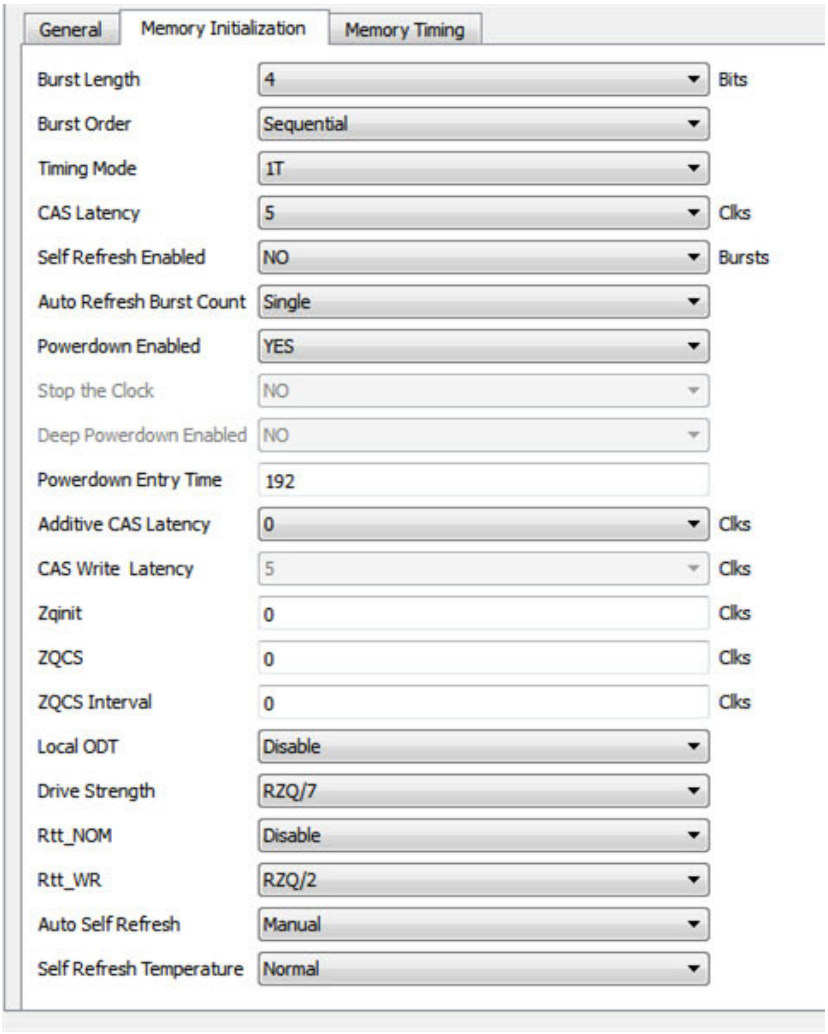

Figure 1-3 **• MDDR Configuration—Memory Initialization Parameters (DDR3)**

Memory Timing

This tab allows you to configure the Memory Timing parameters. Refer to the Data Sheet of your LPDDR/DDR2/DDR3 memory when configuring the Memory Timing parameters.

When you change or enter a value, the Register Description pane gives you the register name and register value that is updated. Invalid values are flagged as warnings.

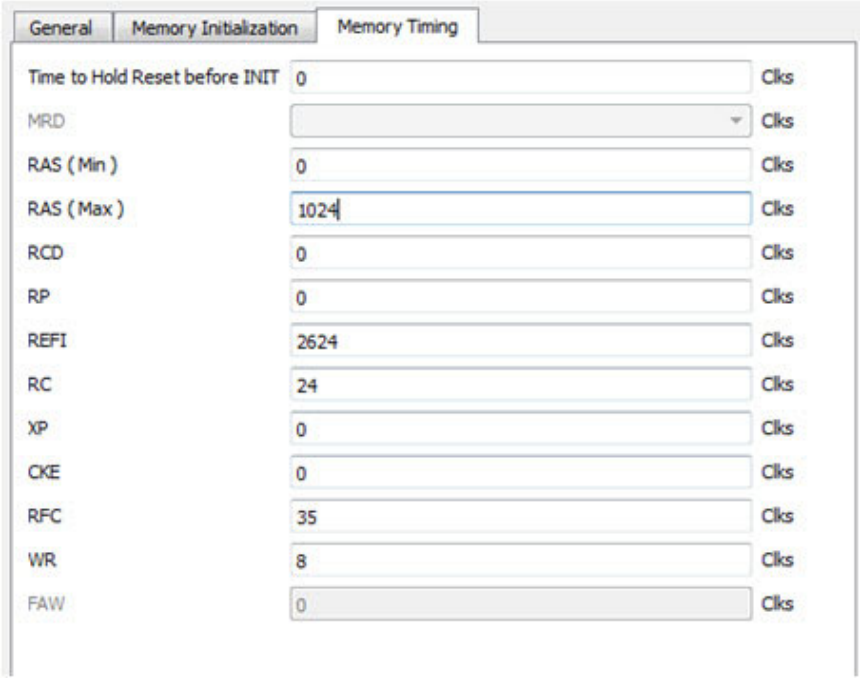

***Figure 1-4* • MDDR Configuration—Memory Timing Tab**

Importing DDR Configuration Files

In addition to entering DDR Memory parameters using the Memory Initialization and Timing tabs, you can import DDR register values from a file. To do so, click **Import Configuration** and navigate to the text file containing DDR register names and values. Figure 1-5 shows the import configuration syntax.

```
ddrc_dyn_soft_reset_CR          0x00 ;
ddrc_dyn_refresh_1_CR           0x27DE ;
ddrc_dyn_refresh_2_CR           0x030F ;
ddrc_dyn_powerdown_CR           0x02 ;
ddrc_dyn_debug_CR               0x00 ;
ddrc_ecc_data_mask_CR           0x0000 ;
ddrc_addr_map_col_1_CR          0x3333 ;
ddrc_addr_map_col_3_CR          0x3300 ;
ddrc_init_1_CR                  0x0001 ;
ddrc_cke_rstn_cycles_CR1        0x0100 ;
ddrc_cke_rstn_cycles_CR2        0x0008 ;
ddrc_init_emr2_CR               0x0000 ;
ddrc_init_emr3_CR               0x0000 ;
ddrc_dram_bank_act_timing_CR    0x1947;
```

***Figure 1-5* • DDR Register Configuration File Syntax**

Note: If you choose to import register values rather than entering them using the GUI, you must specify all necessary register values. For details, refer to the IGLOO2 High Speed DDR Interfaces User Guide.

Exporting DDR Configuration Files

You can also export the current register configuration data into a text file. This file contains register values that you imported (if any), as well as those that were computed from GUI parameters you entered in this dialog box.

If you want to undo changes you have made to the DDR register configuration, you can do so with Restore Default. This deletes all register configuration data and you must either re-import or reenter this data. The data is reset to the hardware reset values.

Generated Data

Click **OK** to generate the configuration. Based on your input in the General, Memory Timing and Memory Initialization tabs, the FDDR Configurator computes values for all DDR configuration registers and exports these values into your firmware project and simulation files. The exported file syntax is shown in Figure 1-6.

```
# Exported: 2013-Sep-02 05:07:16
# Libero DDR Configurator GUI Version = 2.0
# DDR Controller Type = DDR2
# Bus Width = 32-bits
# Memory Bandwidth = 200 Mbps
# Total Bandwidth = 6400 Mbps
#
# Validation Status:
# Target Device Manufacturer:
# Target Device:
#
# User Comments:
#
#
DDRC_ADDR_MAP_BANK_CR.REG_DDRC_ADDRMAP_BANK_B2                    0xa
DDRC_ADDR_MAP_BANK_CR.REG_DDRC_ADDRMAP_BANK_B1                    0xa
DDRC_ADDR_MAP_BANK_CR.REG_DDRC_ADDRMAP_BANK_B0                    0xa
DDRC_ADDR_MAP_COL_1_CR.REG_DDRC_ADDRMAP_COL_B7                    0x3
DDRC_ADDR_MAP_COL_1_CR.REG_DDRC_ADDRMAP_COL_B4                    0x3
DDRC_ADDR_MAP_COL_1_CR.REG_DDRC_ADDRMAP_COL_B3                    0x3
DDRC_ADDR_MAP_COL_1_CR.REG_DDRC_ADDRMAP_COL_B2                    0x3
DDRC_ADDR_MAP_COL_2_CR.REG_DDRC_ADDRMAP_COL_B11                   0xf
DDRC_ADDR_MAP_COL_2_CR.REG_DDRC_ADDRMAP_COL_B10                   0xf
DDRC_ADDR_MAP_COL_2_CR.REG_DDRC_ADDRMAP_COL_B9                    0xf
DDRC_ADDR_MAP_COL_2_CR.REG_DDRC_ADDRMAP_COL_B8                    0x3
DDRC_ADDR_MAP_COL_3_CR.REG_DDRC_ADDRMAP_COL_B6                    0x3
DDRC_ADDR_MAP_COL_3_CR.REG_DDRC_ADDRMAP_COL_B5                    0x3
```

***Figure 1-6* • Exported DDR Register Configuration File Syntax**

HPMS DDR Initialization

The Register Configuration data you import for the HPMS DDR are loaded into the eNVM and copied to the HPMS DDR configuration registers upon FPGA reset. No user action is required to initialize the HPMS DDR at runtime. This automated initialization is also modeled in simulation.

2 – Port Description

DDR PHY Interface

These ports are exposed at the top level of the System Builder generated block. For details, consult the IGLOO2 System Builder User Guide. Connect these ports to your DDR memory.

***Table 2-1* • DDR PHY Interface**

Port Name	Direction	Description	Remarks
MDDR_CAS_N	OUT	DRAM CASN	
MDDR_CKE	OUT	DRAM CKE	
MDDR_CLK	OUT	Clock, P side	
MDDR_CLK_N	OUT	Clock, N side	
MDDR_CS_N	OUT	DRAM CSN	
MDDR_ODT	OUT	DRAM ODT	Ignore this signal for LPDDR Interface. For LPDDR, mark it unused.
MDDR_RAS_N	OUT	DRAM RASN	
MDDR_RESET_N	OUT	DRAM Reset for DDR3	Ignore this signal for LPDDR Interface. For LPDDR, mark it unused.
MDDR_WE_N	OUT	DRAM WEN	
MDDR_ADDR[15:0]	OUT	Dram Address bits	Address MSB vary with the number of rows in the DDR memory. See Table 2-2 below.
MDDR_BA[2:0]	OUT	Dram Bank Address	For LPDDR, BA[2] is not used. Slice the bus and mark BA[2] unused. Connect BA[1:0] to LPDDR.
MDDR_DM_RDQS ([3:0]/[1:0]/[0])	INOUT	Dram Data Mask	For LPDDR interface, this port direction is OUT. RDQS function is not supported. Only DM function is supported. Connect it to DRAM_DM input port of LPDDR.
MDDR_DQS ([3:0]/[1:0]/[0])	INOUT	Dram Data Strobe Input/Output - P Side	
MDDR_DQS_N ([3:0]/[1:0]/[0])	INOUT	Dram Data Strobe Input/Output - N Side	
MDDR_DQ ([31:0]/[15:0]/[7:0])	INOUT	DRAM Data Input/ Output	
MDDR_DQS_TMATCH_0_IN	IN	FIFO in signal	For LPDDR, connect this signal to FDDR_DQS_TMATCH_0_OUT.
MDDR_DQS_TMATCH_0_OUT	OUT	FIFO out signal	For LPDDR, connect this signal to FDDR_DQS_TMATCH_0_IN.
MDDR_DQS_TMATCH_1_IN	IN	FIFO in signal (32-bit only)	For LPDDR, connect this signal to FDDR_DQS_TMATCH_1_OUT.

***Table 2-1* • DDR PHY Interface (continued)**

Port Name	Direction	Description	Remarks
MDDR_DQS_TMATCH_1_OUT	OUT	FIFO out signal (32-bit only)	For LPDDR, connect this signal to FDDR_DQS_TMATCH_1_IN.
MDDR_DM_RDQS_ECC	INOUT	Dram ECC Data Mask	
MDDR_DQS_ECC	INOUT	Dram ECC Data Strobe Input/Output - P Side	
MDDR_DQS_ECC_N	INOUT	Dram ECC Data Strobe Input/Output - N Side	
MDDR_DQ_ECC ([3:0]/[1:0]/[0])	INOUT	DRAM ECC Data Input/Output	
MDDR_DQS_TMATCH_ECC_IN	IN	ECC FIFO in signal	
MDDR_DQS_TMATCH_ECC_OUT	OUT	ECC FIFO out signal (32-bit only)	

Port widths for some ports change depending on the selection of the PHY width. The notation "[a:0]/[b:0]/[c:0]" is used to denote such ports, where "[a:0]" refers to the port width when a 32-bit PHY width is selected, "[b:0]" corresponds to a 16-bit PHY width, and "[c:0]" corresponds to an 8-bit PHY width.

***Table 2-2* • Address MSB Value for LPDDR**

LPDDR Memory Width	64 MB	128 MB	256 MB	512 MB	1 GB	2 GB
16-bit	11	11	12	12	13	13
32-bit	10	11	11	12	12 OR 13	13

Note: This Address MSB table is for individual DDR components. A DDR dim/board may use several identical components to increase the size (for example, 2 x16 2GB to create x32 4GB). In that case, it is the individual component width/depth that controls ADDR MSB. It is also equivalent to the row address width parameter in the configurator.

Fabric Master AXI Bus Interface

***Table 2-3* • Fabric Master AXI Bus Interface**

Port Name	Direction	Description
DDR_AXI_S_AWREADY	OUT	Write address ready
DDR_AXI_S_WREADY	OUT	Write address ready
DDR_AXI_S_BID[3:0]	OUT	Response ID
DDR_AXI_S_BRESP[1:0]	OUT	Write response
DDR_AXI_S_BVALID	OUT	Write response valid
DDR_AXI_S_ARREADY	OUT	Read address ready
DDR_AXI_S_RID[3:0]	OUT	Read ID Tag
DDR_AXI_S_RRESP[1:0]	OUT	Read Response
DDR_AXI_S_RDATA[63:0]	OUT	Read data
DDR_AXI_S_RLAST	OUT	Read Last This signal indicates the last transfer in a read burst
DDR_AXI_S_RVALID	OUT	Read address valid

***Table 2-3* • Fabric Master AXI Bus Interface (continued)**

Port Name	Direction	Description
DDR_AXI_S_AWID[3:0]	IN	Write Address ID
DDR_AXI_S_AWADDR[31:0]	IN	Write address
DDR_AXI_S_AWLEN[3:0]	IN	Burst length
DDR_AXI_S_AWSIZE[1:0]	IN	Burst size
DDR_AXI_S_AWBURST[1:0]	IN	Burst type
DDR_AXI_S_AWLOCK[1:0]	IN	Lock type This signal provides additional information about the atomic characteristics of the transfer
DDR_AXI_S_AWVALID	IN	Write address valid
DDR_AXI_S_WID[3:0]	IN	Write Data ID tag
DDR_AXI_S_WDATA[63:0]	IN	Write data
DDR_AXI_S_WSTRB[7:0]	IN	Write strobes
DDR_AXI_S_WLAST	IN	Write last
DDR_AXI_S_WVALID	IN	Write valid
DDR_AXI_S_BREADY	IN	Write ready
DDR_AXI_S_ARID[3:0]	IN	Read Address ID
DDR_AXI_S_ARADDR[31:0]	IN	Read address
DDR_AXI_S_ARLEN[3:0]	IN	Burst length
DDR_AXI_S_ARSIZE[1:0]	IN	Burst size
DDR_AXI_S_ARBURST[1:0]	IN	Burst type
DDR_AXI_S_ARLOCK[1:0]	IN	Lock Type
DDR_AXI_S_ARVALID	IN	Read address valid
DDR_AXI_S_RREADY	IN	Read address ready
DDR_AXI_S_CORE_RESET_N	IN	MDDR Global Reset
DDR_AXI_S_RMW	IN	Indicates whether all bytes of a 64 bit lane are valid for all beats of an AXI transfer. 0: Indicates that all bytes in all beats are valid in the burst and the controller should default to write commands 1: Indicates that some bytes are invalid and the controller should default to RMW commands This is classed as an AXI write address channel sideband signal and is valid with the AWVALID signal. Only used when ECC is enabled.

Fabric Master AHB0 Bus Interface

***Table 2-4* • Fabric Master AHB0 Bus Interface**

Port Name	Direction	Description
DDR_AHB0_SHREADYOUT	OUT	AHBL slave ready - When high for a write indicates the MDDR is ready to accept data and when high for a read indicates that data is valid
DDR_AHB0_SHRESP	OUT	AHBL response status - When driven high at the end of a transaction indicates that the transaction has completed with errors. When driven low at the end of a transaction indicates that the transaction has completed successfully.
DDR_AHB0_SHRDATA[31:0]	OUT	AHBL read data - Read data from the MDDR slave to the fabric master
DDR_AHB0_SHSEL	IN	AHBL slave select - When asserted, the MDDR is the currently selected AHBL slave on the fabric AHB bus
DDR_AHB0_SHADDR[31:0]	IN	AHBL address - byte address on the AHBL interface
DDR_AHB0_SHBURST[2:0]	IN	AHBL Burst Length
DDR_AHB0_SHSIZE[1:0]	IN	AHBL transfer size - Indicates the size of the current transfer (8/16/32 byte transactions only)
DDR_AHB0_SHTRANS[1:0]	IN	AHBL transfer type - Indicates the transfer type of the current transaction
DDR_AHB0_SHMASTLOCK	IN	AHBL lock - When asserted the current transfer is part of a locked transaction
DDR_AHB0_SHWRITE	IN	AHBL write - When high indicates that the current transaction is a write. When low indicates that the current transaction is a read
DDR_AHB0_S_HREADY	IN	AHBL ready - When high, indicates that the MDDR is ready to accept a new transaction
DDR_AHB0_S_HWDATA[31:0]	IN	AHBL write data - Write data from the fabric master to the MDDR

Fabric Master AHB1 Bus Interface

***Table 2-5* • Fabric Master AHB1 Bus Interface**

Port Name	Direction	Description
DDR_AHB1_SHREADYOUT	OUT	AHBL slave ready - When high for a write indicates the MDDR is ready to accept data and when high for a read indicates that data is valid
DDR_AHB1_SHRESP	OUT	AHBL response status - When driven high at the end of a transaction indicates that the transaction has completed with errors. When driven low at the end of a transaction indicates that the transaction has completed successfully.
DDR_AHB1_SHRDATA[31:0]	OUT	AHBL read data - Read data from the MDDR slave to the fabric master
DDR_AHB1_SHSEL	IN	AHBL slave select - When asserted, the MDDR is the currently selected AHBL slave on the fabric AHB bus
DDR_AHB1_SHADDR[31:0]	IN	AHBL address - byte address on the AHBL interface
DDR_AHB1_SHBURST[2:0]	IN	AHBL Burst Length
DDR_AHB1_SHSIZE[1:0]	IN	AHBL transfer size - Indicates the size of the current transfer (8/16/32 byte transactions only)
DDR_AHB1_SHTRANS[1:0]	IN	AHBL transfer type - Indicates the transfer type of the current transaction
DDR_AHB1_SHMASTLOCK	IN	AHBL lock - When asserted the current transfer is part of a locked transaction
DDR_AHB1_SHWRITE	IN	AHBL write - When high indicates that the current transaction is a write. When low indicates that the current transaction is a read.
DDR_AHB1_SHREADY	IN	AHBL ready - When high, indicates that the MDDR is ready to accept a new transaction
DDR_AHB1_SHWDATA[31:0]	IN	AHBL write data - Write data from the fabric master to the MDDR

Soft Memory Controller Mode AXI Bus Interface

***Table 2-6* • Soft Memory Controller Mode AXI Bus Interface**

Port Name	Direction	Description
SMC_AXI_M_WLAST	OUT	Write last
SMC_AXI_M_WVALID	OUT	Write valid
SMC_AXI_M_AWLEN[3:0]	OUT	Burst length
SMC_AXI_M_AWBURST[1:0]	OUT	Burst type
SMC_AXI_M_BREADY	OUT	Response ready
SMC_AXI_M_AWVALID	OUT	Write Address Valid
SMC_AXI_M_AWID[3:0]	OUT	Write Address ID
SMC_AXI_M_WDATA[63:0]	OUT	Write Data
SMC_AXI_M_ARVALID	OUT	Read address valid
SMC_AXI_M_WID[3:0]	OUT	Write Data ID tag
SMC_AXI_M_WSTRB[7:0]	OUT	Write strobes
SMC_AXI_M_ARID[3:0]	OUT	Read Address ID
SMC_AXI_M_ARADDR[31:0]	OUT	Read address
SMC_AXI_M_ARLEN[3:0]	OUT	Burst length
SMC_AXI_M_ARSIZE[1:0]	OUT	Burst size
SMC_AXI_M_ARBURST[1:0]	OUT	Burst type
SMC_AXI_M_AWADDR[31:0]	OUT	Write Address
SMC_AXI_M_RREADY	OUT	Read address ready
SMC_AXI_M_AWSIZE[1:0]	OUT	Burst size
SMC_AXI_M_AWLOCK[1:0]	OUT	Lock type This signal provides additional information about the atomic characteristics of the transfer
SMC_AXI_M_ARLOCK[1:0]	OUT	Lock Type
SMC_AXI_M_BID[3:0]	IN	Response ID
SMC_AXI_M_RID[3:0]	IN	Read ID Tag
SMC_AXI_M_RRESP[1:0]	IN	Read Response
SMC_AXI_M_BRESP[1:0]	IN	Write response
SMC_AXI_M_AWREADY	IN	Write address ready
SMC_AXI_M_RDATA[63:0]	IN	Read Data
SMC_AXI_M_WREADY	IN	Write ready
SMC_AXI_M_BVALID	IN	Write response valid
SMC_AXI_M_ARREADY	IN	Read address ready
SMC_AXI_M_RLAST	IN	Read Last This signal indicates the last transfer in a read burst
SMC_AXI_M_RVALID	IN	Read Valid

Soft Memory Controller Mode AHB0 Bus Interface

***Table 2-7* • Soft Memory Controller Mode AHB0 Bus Interface**

Port Name	Direction	Description
SMC_AHB_M_HBURST[1:0]	OUT	AHBL Burst Length
SMC_AHB_M_HTRANS[1:0]	OUT	AHBL transfer type - Indicates the transfer type of the current transaction.
SMC_AHB_M_HMASTLOCK	OUT	AHBL lock - When asserted the current transfer is part of a locked transaction
SMC_AHB_M_HWRITE	OUT	AHBL write -- When high indicates that the current transaction is a write. When low indicates that the current transaction is a read
SMC_AHB_M_HSIZE[1:0]	OUT	AHBL transfer size - Indicates the size of the current transfer (8/16/32 byte transactions only)
SMC_AHB_M_HWDATA[31:0]	OUT	AHBL write data - Write data from the MSS master to the fabric Soft Memory Controller
SMC_AHB_M_HADDR[31:0]	OUT	AHBL address - byte address on the AHBL interface
SMC_AHB_M_HRESP	IN	AHBL response status - When driven high at the end of a transaction indicates that the transaction has completed with errors. When driven low at the end of a transaction indicates that the transaction has completed successfully
SMC_AHB_M_HRDATA[31:0]	IN	AHBL read data - Read data from the fabric Soft Memory Controller to the MSS master
SMC_AHB_M_HREADY	IN	AHBL ready - High indicates that the AHBL bus is ready to accept a new transaction

A – Product Support

Microsemi SoC Products Group backs its products with various support services, including Customer Service, Customer Technical Support Center, a website, electronic mail, and worldwide sales offices. This appendix contains information about contacting Microsemi SoC Products Group and using these support services.

Customer Service

Contact Customer Service for non-technical product support, such as product pricing, product upgrades, update information, order status, and authorization.

From North America, call 800.262.1060
From the rest of the world, call 650.318.4460
Fax, from anywhere in the world, 408.643.6913

Customer Technical Support Center

Microsemi SoC Products Group staffs its Customer Technical Support Center with highly skilled engineers who can help answer your hardware, software, and design questions about Microsemi SoC Products. The Customer Technical Support Center spends a great deal of time creating application notes, answers to common design cycle questions, documentation of known issues, and various FAQs. So, before you contact us, please visit our online resources. It is very likely we have already answered your questions.

Technical Support

Visit the Customer Support website (www.microsemi.com/soc/support/search/default.aspx) for more information and support. Many answers available on the searchable web resource include diagrams, illustrations, and links to other resources on the website.

Website

You can browse a variety of technical and non-technical information on the SoC home page, at www.microsemi.com/soc.

Contacting the Customer Technical Support Center

Highly skilled engineers staff the Technical Support Center. The Technical Support Center can be contacted by email or through the Microsemi SoC Products Group website.

Email

You can communicate your technical questions to our email address and receive answers back by email, fax, or phone. Also, if you have design problems, you can email your design files to receive assistance. We constantly monitor the email account throughout the day. When sending your request to us, please be sure to include your full name, company name, and your contact information for efficient processing of your request.

The technical support email address is soc_tech@microsemi.com.

My Cases

Microsemi SoC Products Group customers may submit and track technical cases online by going to My Cases.

Outside the U.S.

Customers needing assistance outside the US time zones can either contact technical support via email (soc_tech@microsemi.com) or contact a local sales office. Sales office listings can be found at www.microsemi.com/soc/company/contact/default.aspx.

ITAR Technical Support

For technical support on RH and RT FPGAs that are regulated by International Traffic in Arms Regulations (ITAR), contact us via soc_tech_itar@microsemi.com. Alternatively, within My Cases, select **Yes** in the ITAR drop-down list. For a complete list of ITAR-regulated Microsemi FPGAs, visit the ITAR web page.

Microsemi Corporate Headquarters
One Enterprise, Aliso Viejo,
CA 92656 USA

Within the USA: +1 (800) 713-4113
Outside the USA: +1 (949) 380-6100
Sales: +1 (949) 380-6136
Fax: +1 (949) 215-4996

E-mail: sales.support@microsemi.com

Microsemi Corporation (MSCC) offers a comprehensive portfolio of semiconductor and system solutions for communications, defense & security, aerospace, and industrial markets. Products include high-performance and radiation-hardened analog mixed-signal integrated circuits, FPGAs, SoCs, and ASICs; power management products; timing and synchronization devices and precise time solutions, setting the world's standard for time; voice processing devices; RF solutions; discrete components; security technologies and scalable anti-tamper products; Ethernet solutions; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Microsemi is headquartered in Aliso Viejo, Calif. and has approximately 3,600 employees globally. Learn more at **www.microsemi.com**.