APPLICATION NOTE

ARMADA 16X Memory Controller Configuration and Tuning

1. Overview

The ARMADA 16x Memory Controller Unit (MCU) supports LPDDR1, DDR2, and LVDDR2. It does not support LPDDR2. It is critical that the MCU be programmed correctly to ensure system stability.

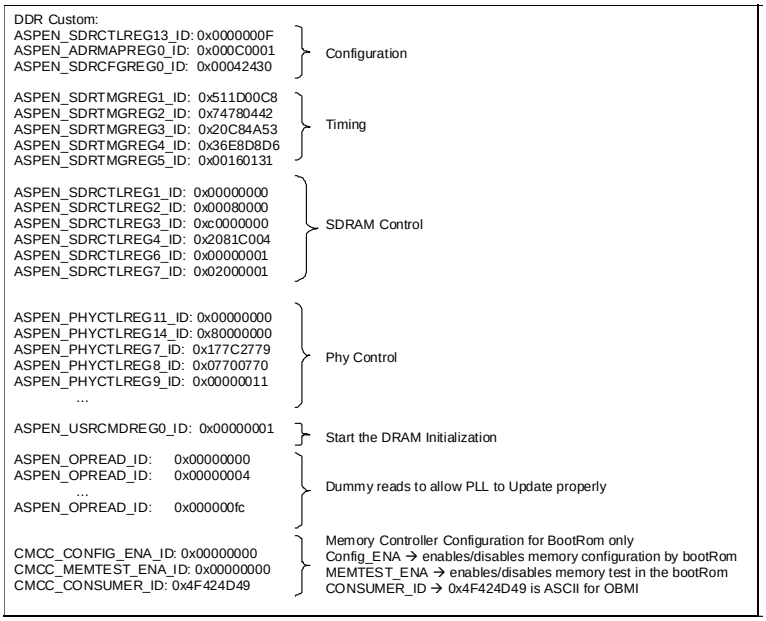
2. Programming Sequence.

The *ARMADA 16x Applications Processor Family Software Manual* (MV-S301544-00) lists the order in which the MCU registers must be programmed. The Marvell OEM Boot Module (OBM) contains sample code that sets up the MCU.

3. NTIM and OBM

On bootup, the Boot ROM loads the OBM from storage memory to internal SRAM and hands control to it. The OBM reads content of the Non-Trusted Image Module (NTIM) and searches for a DDR configuration package. If one exists, the OBM uses the information in the NTIM to configure the MCU. The NTIM can be customized for specific DDR devices by changing the register values. The sequence in which these registers are arranged in the NTIM description file is the same sequence in which these registers are programmed by OBM. The following is an example section of the NTIM Descriptor file that relates to DDR configuration.

Figure 1: Section of NTIM Description File Concerning DDR

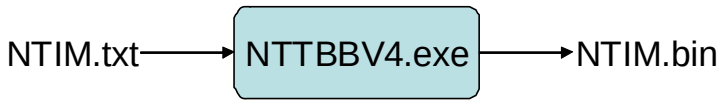


An NTIM binary can be generated from the NTIM description file using the latest WTPTP release package available on the Marvell extranet at:

My Products/Cellular & Handheld Solutions/Applications Processors/ARMADA 16x (ARMADA 16x - 88AP16X)/ Software > WTPTP

Specifically, the WTPTP NTTBBV4.exe tool is used to generate the NTIM binary (see Figure 2).

Figure 2: Generating the NTIM binary



Alternatively, an existing NTIM binary can be modified using a hex editor. Figure 3: "Example DDR Package within the NTIM Binary File" shows the DDR information in the NTIM binary. Starting at address 0x110 (in Figure 3) is a list of ID/value MCU register pairs. For convenience, the registers IDs are shown on the right side of Figure 3.

Ensure the file size is not altered if editing the binary NTIM file..

Figure 3: Example DDR Package within the NTIM Binary File

NTIM and OBM are OS-agnostic. OBM can be compiled on a Windows XP machine or on a Linux machine. To compile OBM, the Software Development Suite (SDT) is needed and is available here:

My Products/Cellular & Handheld Solutions/Development Tools/General/C++ SDT

The following sections explain how to arrive at the register values.

4. Configuration Registers

SDRAM_CNTRL_13

MEMORY_ADDRESS_MAP_REG0

SDRAM_CONFIG_0

These registers configure the size of the DDR device (that is, 128MB, 256MB), number of banks, number of rows and columns as well as drive strength and on-die termination. Partial self-refresh for LPDDR1 can also be configured.

The Excel spreadsheet that complements this app note can be used to calculate registers SDRAM_CONFIG_0 and MEMORY_ADDRESS_MAP_REG0.

The specification sheet of a DDR device contains the necessary information to configure these registers.

5. Timing Registers

SDRAM_TIMING_1/2/3/4/5

The specification sheet of a DDR device contains the timing values needed for proper operation. These values typically are listed in nanoseconds (ns) or number of data clock cycles (tCK). The ARMADA 16x MCU accepts those timings in number of data clocks only, so conversion is necessary if specified in fractions of a second.

Example:

At 200 MHz clock, one data clock cycle = 5 ns

At 400 MHz clock, one data clock cycle = 2.5 ns

So if the spec requires tRTP to be a minimum of 7.5 ns, then

At 200 MHz clock, program the SDRAM_Config0[tRTP] with 2

At 400 MHz clock, program the SDRAM_Config0[tRTP] with 3

The Excel spreadsheet that complements this app note can be used to calculate the values for all the timing registers. The following three diagrams show the meaning of the timing fields used by the MCU in the ARMADA 16X Applications Processor.

Figure 4: Timing Diagram 1

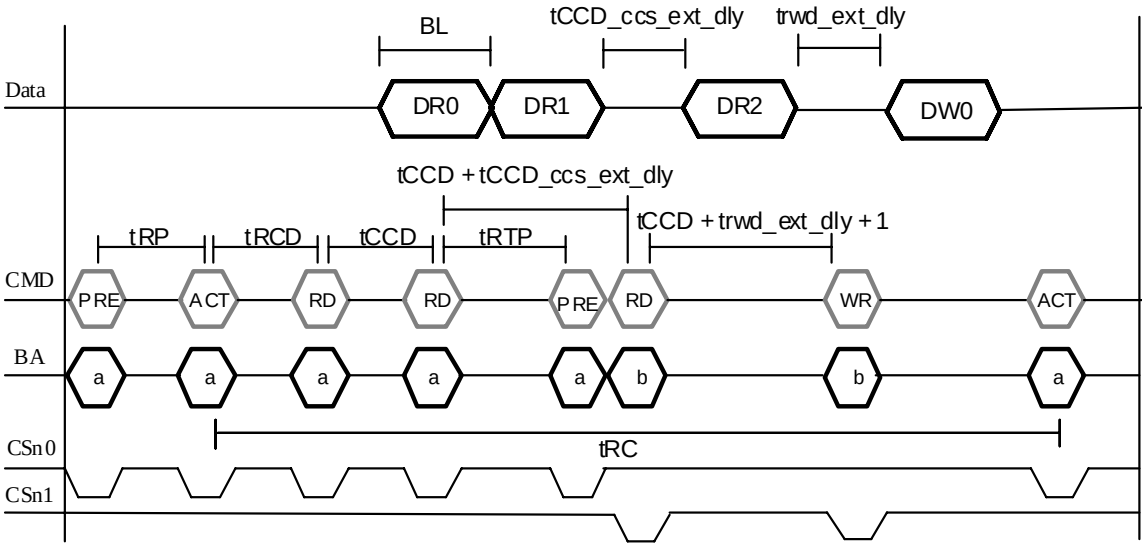


Figure 5: Timing Diagram 2

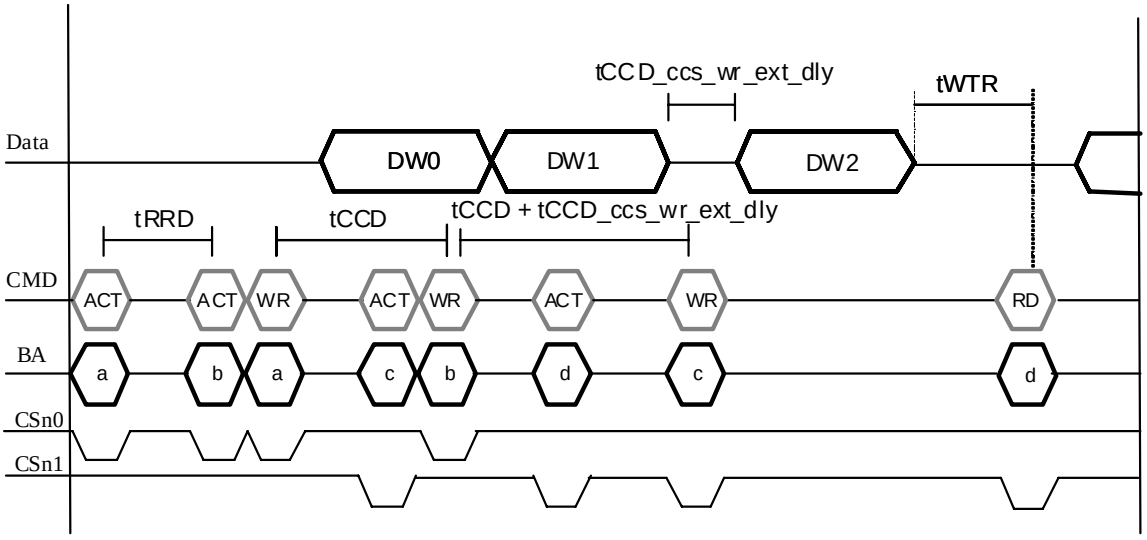
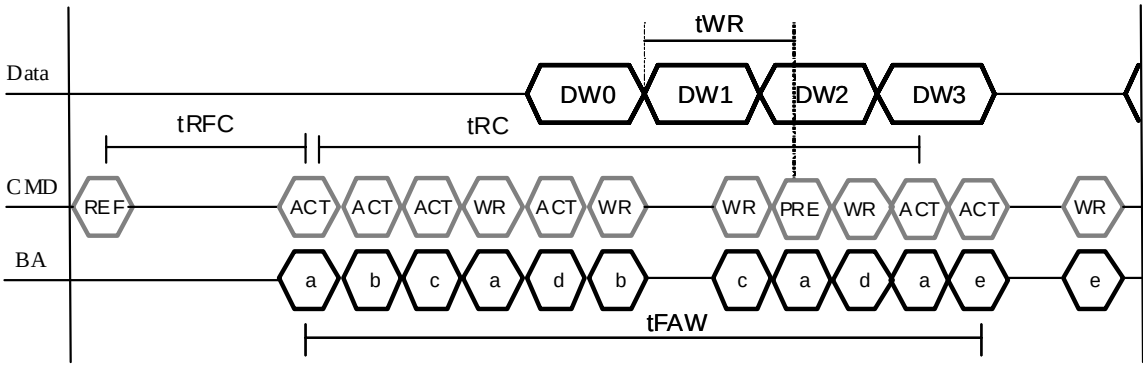


Figure 6: Timing Diagram 3



6. Control Registers

SDRAM_CNTRL_1/2/3/4/5/6/7

These registers control automatic power saving, automatic clock stopping (LPDDR1 only), DLL resetting (DDR2 only), SDRAM line boundary (set to line_64), auto-precharging, Powerdown mode (DDR2 only), refresh posting enabling, fast and early write enabling, data width on the DDR device, SDRAM type (DDR1 or DDR2), CAS latency, burst length, fast/normal bank addressing, arbiter master weights for the four masters, pad termination mode for Reads, and On-die termination for Read and Write operations.

Refer to the *ARMADA 16x Applications Processor Hardware Manual* (MV-S301545-00) for details.

The Excel spreadsheet that complements this app note can be used to determine the register values for the SDRAM_CNTRL_4 and SDRAM_CNTRL_7.

7. PHY Registers and Tuning

PHY_CNTRL_3/7/8/9/10/11/12/13/14

The ARMADA 16x DDR PHY is a universal PHY (U-PHY). Adjust the driver strength and termination for DQ/DQS (data and data strobe), ADCM (Address and command), and CK (clock) to obtain optimal functionality. Tune the DDR PHY for either signal integrity (recommended) or for low power.

7.1 Optimal Signal Integrity Tuning

This approach adjusts for optimal and quantifiable signal integrity using an oscilloscope.

The goal is to have as steep as possible signal rise and fall times on the ARMADA 16x side with no under- or overswing on the signal. This procedure must be verified with a high-speed capable (active or passive) scope probe.

In the simplest case, connect a 50-ohm coax cable to the scope with 50-ohm termination. On the other end, connect the cable to the signal of interest via a 500- or 1000-ohm resistor. The ARMADA 16x drive strength is set up in a similar fashion but with the scope probe on the memory side of the signal. The signal strength is increased until under- and over-swing is observed.

Each of the signals to the DDR memory can be adjusted to match the impedance of the transmission lines or traces to the DDR memory. To compensate for voltage and temperature, an external resistor is used to calibrate the driver impedance.

Each input signal from the DDR to the PHY can be terminated internally within the PHY. The impedance of the termination is adjustable to match the source impedance of the DDR.

Note

Since the DQ and the DQS lines provide on-die termination (ODT), no external series or parallel termination is required on those lines for point-to-point connections. This condition would apply to systems that use only one DDR memory chip. The CK, CK#, and address and control lines require parallel termination on the DDR memory side (75 ohm typically).

7.2 Low Power Tuning

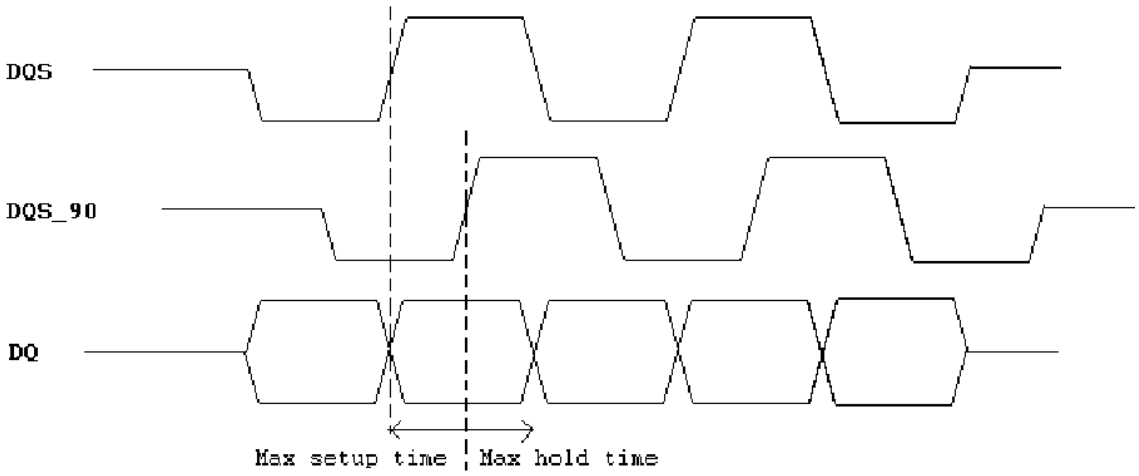
The goal here is to use minimal drive strength and thus power consumption at the possible expense of signal integrity, which can be performed without a scope. Start with a known working drive setting on the ARMADA 16x and memory side. After writing a pattern to memory, the DDR Controller is re-initialized with lower memory drive strength and the data is read back and checked for errors. If no errors occur, the drive strength can be reduced further. This procedure is repeated until Read failure occurs. When this level of drive has been reached on the memory side, the drive strength must be increased to the next higher level.

Starting again from a known working drive-strength setting on the ARMADA 16x and memory side, the drive strength on the ARMADA 16x side is reduced one step at a time. Next, a pattern is written to memory and read back to check for errors. The drive strength is reduced until errors occur. At this point, the drive strength must be increased by one step. The signal margin obtained by this tuning approach is harder to quantify than using the scope-tuning approach.

7.3 Read DLL

The DLL and the associated delay chain are used to delay the input data strobe (DQS) so that it aligns at the center of the data (DQ), resulting in a maximally balanced setup and hold time. Ninety-degrees phase shift is recommended but values between 45 to 135 degrees can be used.

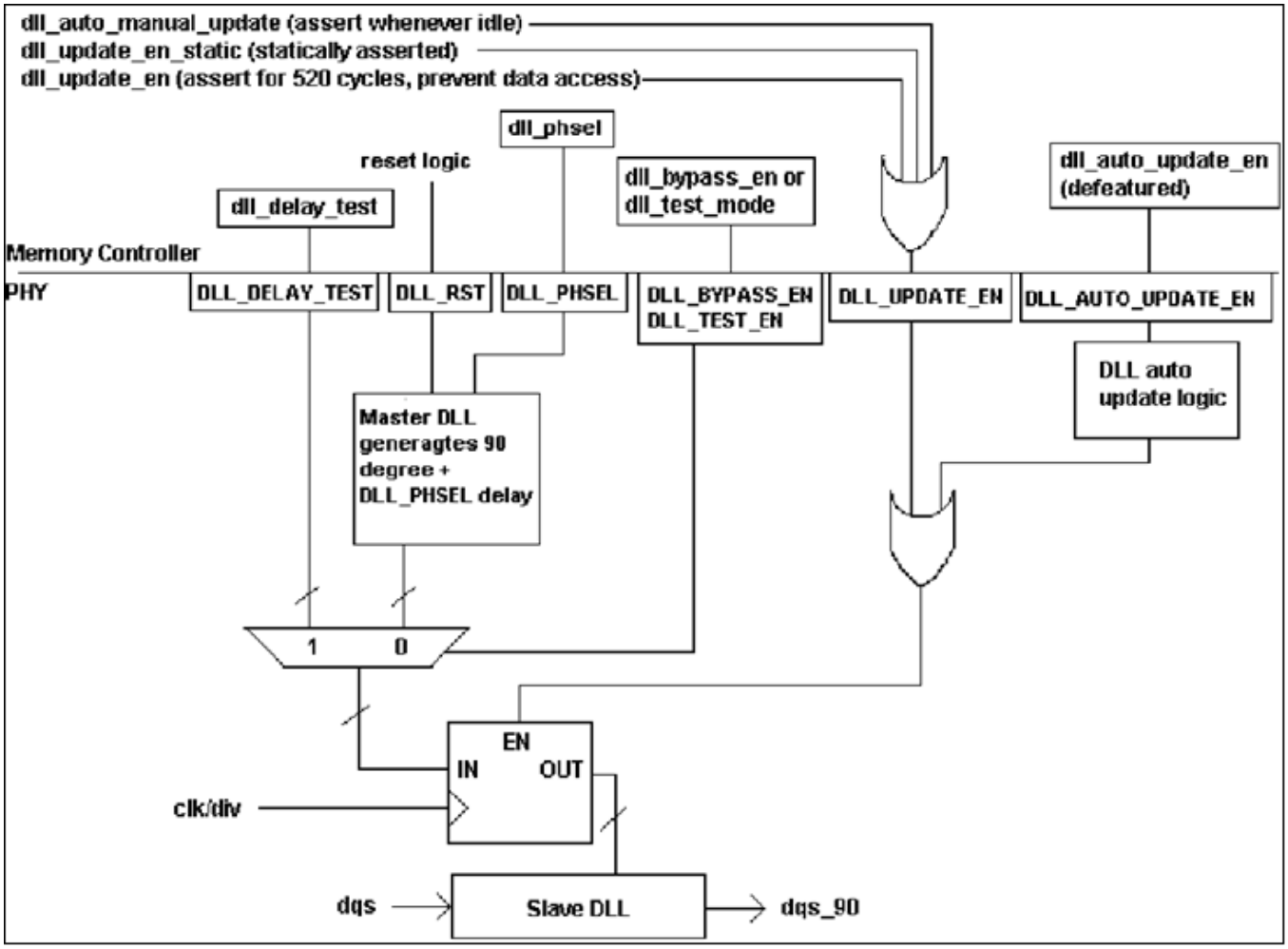
Figure 7: Phase Shifting DQS by 90 Degrees



To calibrate the DLLs, it is essential that the preferred DDR clock speed be set and stabilized before the calibration is enabled. The master DLL is continuously calibrating itself to determine the proper delay value for a 90-degree phase shift. The result of the master DLL calibration is provided in PHY_CNTRL_14[15:8]. If PHY_CNTRL_13[Dll_auto_manual_update_en] is set then the master DLL auto calibration delay value is used, along with the preferred phase values in PHY_CNTRL_13 and

PHY_DLL_CNTRL_1, to control/update the two slave DQS DLLs (DQ[7:0] and DQ[15:0]). This update occurs only when the Memory Controller has been idle for 512 cycles (see Figure 8). This is the safest “auto” method, but periodic updates are not guaranteed, especially during heavy traffic.

Figure 8: DLL Control Diagram

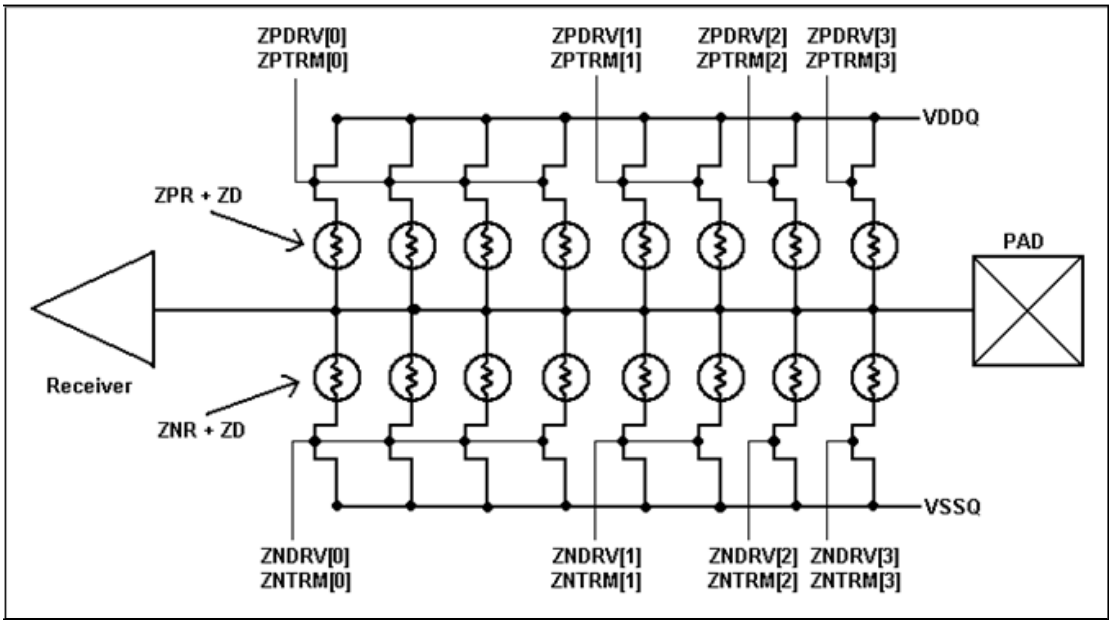


7.4 Drive Strength and Pad Termination Control on ARMADA 16x Devices

LPDDR1 devices typically provide drive strengths of normal, half, quarter, and eight. DDR2 devices provide drive strengths of full or reduced. This drive strength is set up in SDRAM_CONFIG_0[ODS] and communicated to the memory device during initialization of the DDR Memory Controller (see latest DDR setup example code).

ZPDRV and ZNDRV control the PMOS and NMOS coarse drive strength, and are used when the output enable (OE) is enabled, which means the PHY is driving this signal. During this time, only one side (either PMOS or NMOS) is driving at any given time. There are eight total mini-drivers (fingers) on each side. Bit 0 enables four fingers, Bit 1 enables two, and Bits 3 and 2 both enable one each. Thus, drive strength does not increase linearly with ZPDRV and ZNDRV.

Figure 9: Pad Termination and Drive Control



SDRAM_CNTRL_7[pad_term_switch_mode] controls whether pad termination is on or off. When on, ZPTRM and ZNTRM in PHY_CNTRL_7 and PHY_CNTRL_8 control the pad termination strength. When the output enable (OE) is disabled, the drivers on both sides (ARMADA 16x and DDR device) are turned on according to ZPTRM and ZNTRM, resulting in a parallel termination. The finger control scheme is similar to ZPDRV and ZNDRV.

ZPR and ZNR are fine-tuning drivers. The higher the value, the lower the resistance of each finger (stronger drive strength). ZD is added on top of ZPR and ZNR, resulting in a range of 0 (equivalent to three-state, ZD=0 and ZPR/ZNR=0) to 16 (maximum per-finger strength, ZD=1 and ZPR/ZNR=15). Higher values are recommended for higher loads.

Data and data strobe (DQ/DQS), address/command (ADCM), and clock (CK) all have separate controls. CK and ADCM share the same Z*DRV and Z*TRM values. Table 1 maps the signal controls to the correct PHY registers.

Table 1: Mapping Signal Drive and Termination Strength to PHY Registers

	Z*DRV	Z*TRM	ZPR	ZNR	ZD
PHY_CNTRL_7	DQ/DQS	DQ/DQS	DQ/DQS	DQ/DQS	DQ/DQS
PHY_CNTRL_8	ADCM/CK	ADCM/CK	ADCM	ADCM	ADCM
PHY_CNTRL_9	--	--	CK	CK	CK

Table 2: Example Drive Strengths

DDR Type	ZPDRV	ZNDRV	ZPTRM	ZNTRM
Impedance Match (~50 ohms driving, no termination)	0011	0011	0	0
DDR1 (~30 ohms driving, termination on PCB)	1111	1111	0	0
DDR2 (~35 ohms driving, on-die 150 ohms termination to VDDQ and GND)	0111	0111	1100	10

7.5 On-Die Termination

On-Die Termination (ODT) is used with DDR2 and not with LPDDR1. For LPDDR1, the ARMADA 16x ODT must be switched off.

For LPDDR1, ODT must be off:

SDRAM_CONFIG_0[RTT] = 0,

SDRAM_CNTRL_7[ODT0_switch_mode] = 0 (for Chip Select 0)

SDRAM_CNTRL_7[ODT1_switch_mode] = 0 (for Chip Select 1)

For DDR2, ODT must be on:

SDRAM_CONFIG_0[RTT] = see Spec sheet for specific DDR2 device

SDRAM_CNTRL_7[ODT0_switch_mode] = 0x1 or 0x2 (for Chip Select 0)

SDRAM_CNTRL_7[ODT1_switch_mode] = 0x1 or 0x2 (for Chip Select 1)

The RTT value is communicated to the DDR device during initialization.

An ODT0_switch_mode = 0x1 means ODT is enabled for all Reads and Writes (recommended).

An ODT0_switch_mode = 0x2 means ODT is by SDRAM_CNTRL_6[ODT0_write_en] and SDRAM_CNTRL_6[ODT0_read_en]. This mode typically is used in a multi-chip-select system.

7.6 Receiver Type and Voltage Reference

This reference applies only to DQ and DQS lines, since the clock, the address and command, and mode pins are outputs only. Three options exist: internal reference, external reference, and DQSb. The external reference voltage is a good default choice. For receiver type, the CMOS receiver can be used as default for both DQS and DQ. However, the differential provides the best signal integrity at high speed. When using the differential receiver, the external reference voltage should be used for DQ and DQS# as reference voltage for DQS.

8. Entering and Exiting Self-Refresh Mode

Entering Self-Refresh mode:

1. Set PMUA_MC_SW_SLP_TYPE[MC_SW_SLP_TYPE] = 0 to select Self-Refresh as the sleep type of the Memory Controller.
2. Set PMUA_MC_SLP_REQ[MC_SLP_REQ_MOH] = 1 to request entering Self-Refresh mode.
3. Poll PMUA_MC_SLP_REQ[MC_SLP_REQ_ACK] until this bit is set. This confirms the Memory Controller is in Self-Refresh mode.

Exiting Self-Refresh mode:

1. Set PHY_CNTRL_14[Phy_sync_en].
2. (DDR2 only) set SDRAM_CNTRL_1[DLL_reset] to reset DRAM DLL (requires mode-register update).
3. (DDR2 only) set USER_INITIATED_COMMAND[user_Imr0_req] to send MRS command.
4. (DDR2 only) Clear SDRAM_CNTRL_1[DLL_reset].
5. Set PHY_CNTRL_13[dll_reset_timer] = 0x8 (2048 cycles).
6. PHY_CNTRL_14[Phy_DLL_RST] to reset the PHY DLL. This blocks access to the controller for 2048 cycles.
7. Set PHY_CNTRL_14[update_en_static] and wait at least 512 DCLK cycles. This forces an update to the DQS slave DLLs.
8. Set PHY_CNTRL_14[update_en_static] again (this is a 1-to-toggle bit).
9. Set PMUA_MC_SW_SLP_TYPE[MC_SW_SLP_TYPE] = 0 to select Self-Refresh as the sleep type of the Memory Controller.
10. Set PMUA_MC_SLP_REQ[MC_SLP_REQ_MOH] = 0 to request exist from Self-Refresh mode.

11. Poll PMUA_MC_SLP_REQ[MC_SLP_REQ_ACK] until this bit is cleared. This confirms the Memory Controller is out of low-power mode.

9. DDR Testing

The OBM does not contain test code. Marvell suggests developing code that tests for the following:

- Connectivity
- Memory retention
- Stress
- Performance

Marvell Semiconductor, Inc.
5488 Marvell Lane
Santa Clara, CA 95054, USA
Tel: 1.408.222.2500
Fax: 1.408.752.9028
Email: commsales@marvell.com
www.marvell.com