# AM335x DDR PHY register configuration for DDR3 using Software Leveling

This wiki describes how to configure AM335x DDR PHY configuration registers for use with DDR3. These registers need to be configured based on certain circuit board characteristics to ensure proper timing is performed for all DDR transactions. Outlined below is a Software Leveling procedure which should be used as a workaround due to the errata concerning Hardware Leveling in AM335x.

Note: If you are using DDR2 or mDDR, please refer back to the EMIF Configuration Tips (DDR PHY Section) [1] for details on configuration for these types of memories. The procedure below should only be used for DDR3 configuration

## DDR PHY registers

The DDR PHY I/Os are grouped into 5 different sets based on the I/O macro that controls them. Each macro controls 10 signals each (some signals in the each set are unused on AM335x). Three command macros (CMD0-2) contain 10 control/address signals each, and two data macros (DATA0-1) correspond to the 2 data bytes, including the associated data mask signal (ddr_dqmX) and differential data strobe signals (ddr_dqsX, ddr_dqsnX).

A summary of the DDR PHY registers that need to be configured is listed below. All other DDR PHY registers that are not listed can be left in their default state. For CMDx, x is 0,1,2. For DATAx, x is 0 or 1. In all cases, program the same value for each iteration of the macro.

- CMDx_PHY_CTRL_SLAVE_RATIO - use the value calculated in the Ratio Seed spreadsheet
- CMDx_PHY_INVERT_CLKOUT - In addition to programming these registers with 0 or 1 as defined below, plug this value into the spreadsheet to get the proper seed values for the program below.
  - If (DDR_CK length) < (DDR_DQS length), program this register to 1
  - If (DDR_CK length) > (DDR_DQS length), program this register to 0
- DATAx_PHY_RD_DQS_SLAVE_RATIO - use the procedure below to determine the value for these registers
- DATAx_PHY_WR_DQS_SLAVE_RATIO - use the procedure below to determine the value for these registers
- DATAx_PHY_FIFO_WE_SLAVE_RATIO - use the procedure below to determine the value for these registers
- DATAx_PHY_WR_DATA_SLAVE_RATIO - use the procedure below to determine the value for these registers

## Procedure for determining DDR PHY register values

**Files needed for this procedure:**
*Code Composer GEL file*: You can use the AM3358 StarterKit GEL to start with.
*Executable CCS .out file*: Media:DDR3_slave_ratio_search_auto.zip
1. Configure the EMIF timing registers for your DDR. The AM335x EMIF Configuration Tips wiki (AC timing registers section) [2] has information on how to determine these values. Once these values have been calculated, use these values to change the following section of the GEL you are using:

```
//******************************************************************
//EMIF parameters
//******************************************************************
#define DDR2_READ_LATENCY    0x05          //RD_Latency = (CL + 2) - 1
#define DDR2_SDRAM_TIMING1   0x0666B3C9
#define DDR2_SDRAM_TIMING2   0x243631CA
#define DDR2_SDRAM_TIMING3   0x00000337
```

```
#define DDR2_SDRAM_CONFIG    0x41805332          //termination = 1 (75ohm)
                                                  //SDRAM drive = 0 (full)
                                                  //CL = 4 (CAS latency = 4)
                                                  //ROWSIZE = 6 (15 row bits)
                                                  //PAGESIZE = 2 (10 column bits)
#define DDR2_REF_CTRL        0x0000081a    //266*7.8us = 2074.8 = 0x81A
```

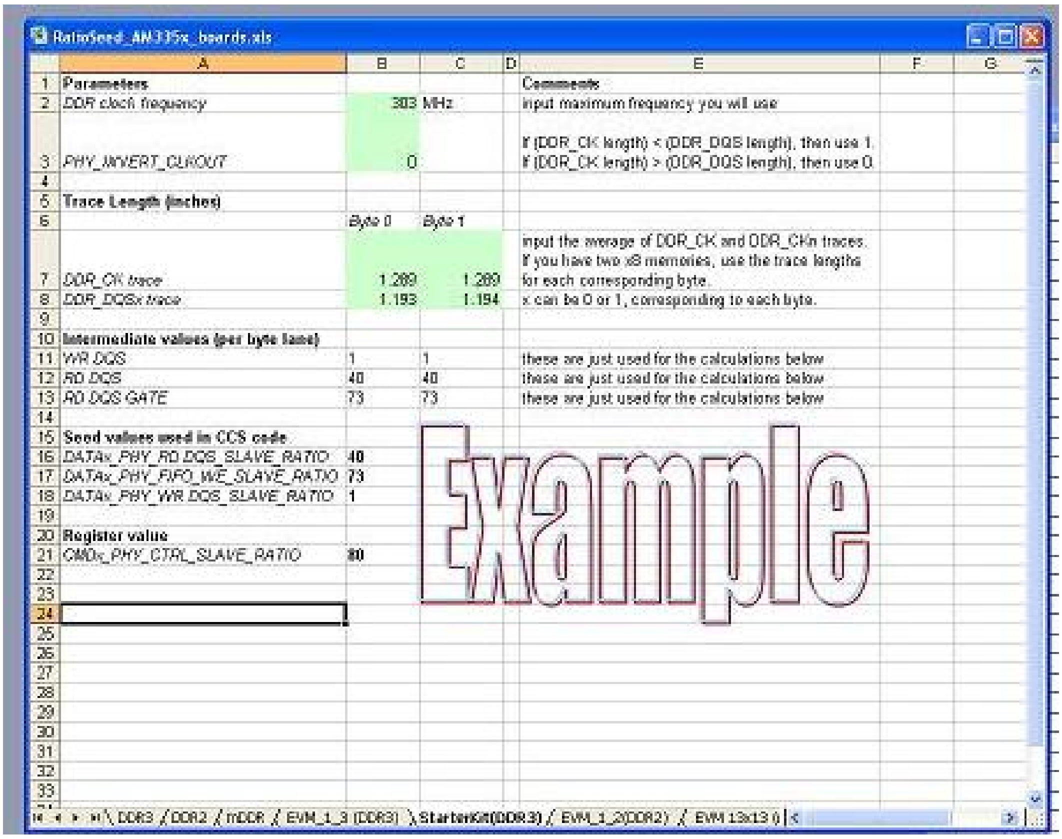
2. Determine the initial seed values to input in the program by using the Ratio Seed spreadsheet.

**Ratio Seed Spreadsheet**

- Choose the 'DDR3' tab at the bottom of the spreadsheet.
- Input the maximum DDR frequency that will be used,
- Enter the appropriate value for PHY_INVERT_CLKOUT (0 or 1)
- Enter the trace length (in inches) of each trace. For DDR_CK, this trace is typically in a 'T' configuration for designs with two x8 memories. Ensure that you input the trace length from AM335x to each memory. These lengths should be close to equal if correct design guidelines were met for a 'T' configuration. For fly-by topology, where the trace runs from AM335x to the first memory, and then to the second memory, ensure that you input the total trace length for each byte.

The spreadsheet also has other tabs that provide examples from other TI development boards. These are to be use only as examples and not for other designs.

Once you finish inputting the appropriate values, you will be using the calculated "Seed Values" at the bottom of the spreadsheet in subsequent steps. Here is an example of the spreadsheet for the StarterKit EVM:



RatioSeed_AM335x_boards.xls

| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | **Parameters** | | | | **Comments** | | |
| 2 | *DDR clock frequency* | 303 | MHz | | input maximum frequency you will use | | |
| 3 | *PHY_INVERT_CLKOUT* | 0 | | | If (DDR_CK length) < (DDR_DQS length), then use 1.<br>If (DDR_CK length) > (DDR_DQS length), then use 0. | | |
| 4 | | | | | | | |
| 5 | **Trace Length (inches)** | | | | | | |
| 6 | | *Byte 0* | *Byte 1* | | | | |
| 7 | *DDR_CK trace* | 1.289 | 1.289 | | input the average of DDR_CK and DDR_CKn traces.<br>If you have two x8 memories, use the trace lengths for each corresponding byte. | | |
| 8 | *DDR_DQSx trace* | 1.193 | 1.194 | | x can be 0 or 1, corresponding to each byte. | | |
| 9 | | | | | | | |
| 10 | **Intermediate values (per byte lane)** | | | | | | |
| 11 | *WR DQS* | 1 | 1 | | these are just used for the calculations below | | |
| 12 | *RD DQS* | 40 | 40 | | these are just used for the calculations below | | |
| 13 | *RD DQS GATE* | 73 | 73 | | these are just used for the calculations below | | |
| 14 | | | | | | | |
| 15 | **Seed values used in CCS code** | | | | | | |
| 16 | *DATAx_PHY_RD_DQS_SLAVE_RATIO* | **40** | | | | | |
| 17 | *DATAx_PHY_FIFO_WE_SLAVE_RATIO* | 73 | | | | | |
| 18 | *DATAx_PHY_WR_DQS_SLAVE_RATIO* | 1 | | | | | |
| 19 | | | | | | | |
| 20 | **Register value** | | | | | | |
| 21 | *CMDx_PHY_CTRL_SLAVE_RATIO* | **80** | | | | | |
| 22 | | | | | | | |
| 23 | | | | | | | |
| 24 | | | | | | | |
| 25 | | | | | | | |
| 26 | | | | | | | |
| 27 | | | | | | | |
| 28 | | | | | | | |
| 29 | | | | | | | |
| 30 | | | | | | | |
| 31 | | | | | | | |
| 32 | | | | | | | |
| 33 | | | | | | | |

3. Connect the board to Code Composer and run the GEL file Scripts->AM335x System Initialization. Load the DDR3_slave_ratio_search_auto.out file from above and run the code. The console window in CCS will prompt you for some input values. Input each value from the spreadsheet (input in hex with no leading '0x') corresponding to the prompt. The program may take a some time to run (especially if using XDS100), as it may iterate in a loop several

times to come up to the optimal values. The console window will show progress and will look something like this:

```
[CortxA8]
[CortxA8] Enter the Seed WR_DQS_SLAVE_RATIO Write DQS Ratio Value in Hex to search the Write DQS Ratio Window
[CortxA8]
[CortxA8] Enter the Seed RD_DQS_SLAVE_RATIO Value in Hex to search the RD DQS Ratio Window
40
[CortxA8]
[CortxA8] Enter the Seed FIFO_WE_SLAVE_RATIO Value in Hex to search the RD DQS Gate Window
F3
[CortxA8]
[CortxA8] Enter the Seed WR_DQS_SLAVE_RATIO Write DQS Ratio Value in Hex to search the Write DQS Ratio Window
81
[CortxA8]
[CortxA8] ***************************************************************
[CortxA8]       The Slave Ratio Search Program Values are...
[CortxA8] ***************************************************************
[CortxA8] PARAMETER                       MAX  |  MIN  | OPTIMUM |  RANGE
[CortxA8] ***************************************************************
[CortxA8] DATA_PHY_RD_DQS_SLAVE_RATIO    0x06f | 0x005 |  0x03a  | 0x06a
[CortxA8] DATA_PHY_FIFO_WE_SLAVE_RATIO   0x1e3 | 0x062 |  0x122  | 0x181
[CortxA8] DATA_PHY_WR_DQS_SLAVE_RATIO    0x0bd | 0x04d |  0x085  | 0x070
[CortxA8] DATA_PHY_WR_DATA_SLAVE_RATIO   0x0f6 | 0x086 |  0x0be  | 0x070
[CortxA8] ***************************************************************
[CortxA8] rd_dqs_range = 3a
[CortxA8] fifo_we_range = 122
[CortxA8] wr_dqs_range = 85
[CortxA8] wr_data_range = be
[CortxA8]
[CortxA8] Optimal values not reached, rerunning program with new values...
[CortxA8]
[CortxA8] ***************************************************************
[CortxA8]       The Slave Ratio Search Program Values are...
[CortxA8] ***************************************************************
[CortxA8] PARAMETER                       MAX  |  MIN  | OPTIMUM |  RANGE
[CortxA8] ***************************************************************
[CortxA8] DATA_PHY_RD_DQS_SLAVE_RATIO    0x06f | 0x005 |  0x03a  | 0x06a
[CortxA8] DATA_PHY_FIFO_WE_SLAVE_RATIO   0x1de | 0x05b |  0x11c  | 0x183
[CortxA8] DATA_PHY_WR_DQS_SLAVE_RATIO    0x0c1 | 0x051 |  0x089  | 0x070
[CortxA8] DATA_PHY_WR_DATA_SLAVE_RATIO   0x0f8 | 0x08a |  0x0c1  | 0x06e
[CortxA8] ***************************************************************
[CortxA8] rd_dqs_range = 0
[CortxA8] fifo_we_range = 6
[CortxA8] wr_dqs_range = 4
[CortxA8] wr_data_range = 3
[CortxA8]
[CortxA8] Optimal values not reached, rerunning program with new values...
[CortxA8]
```

```
[CortxA8] ***************************************************************
[CortxA8]       The Slave Ratio Search Program Values are...
[CortxA8] ***************************************************************
[CortxA8] PARAMETER                       MAX  |  MIN  | OPTIMUM |  RANGE
[CortxA8] ***************************************************************
[CortxA8] DATA_PHY_RD_DQS_SLAVE_RATIO    0x06f | 0x005 |  0x03a  | 0x06a
[CortxA8] DATA_PHY_FIFO_WE_SLAVE_RATIO   0x1dc | 0x05b |  0x11b  | 0x181
[CortxA8] DATA_PHY_WR_DQS_SLAVE_RATIO    0x0c5 | 0x055 |  0x08d  | 0x070
[CortxA8] DATA_PHY_WR_DATA_SLAVE_RATIO   0x0fc | 0x08c |  0x0c4  | 0x070
[CortxA8] ***************************************************************
[CortxA8] rd_dqs_range = 0
[CortxA8] fifo_we_range = 1
[CortxA8] wr_dqs_range = 4
[CortxA8] wr_data_range = 3
[CortxA8]
[CortxA8] Optimal values not reached, rerunning program with new values...
[CortxA8]
[CortxA8] ***************************************************************
[CortxA8]       The Slave Ratio Search Program Values are...
[CortxA8] ***************************************************************
[CortxA8] PARAMETER                       MAX  |  MIN  | OPTIMUM |  RANGE
[CortxA8] ***************************************************************
[CortxA8] DATA_PHY_RD_DQS_SLAVE_RATIO    0x06f | 0x005 |  0x03a  | 0x06a
[CortxA8] DATA_PHY_FIFO_WE_SLAVE_RATIO   0x1de | 0x05b |  0x11c  | 0x183
[CortxA8] DATA_PHY_WR_DQS_SLAVE_RATIO    0x0c8 | 0x05a |  0x091  | 0x06e
[CortxA8] DATA_PHY_WR_DATA_SLAVE_RATIO   0x0ff | 0x091 |  0x0c8  | 0x06e
[CortxA8] ***************************************************************
[CortxA8] rd_dqs_range = 0
[CortxA8] fifo_we_range = 1
[CortxA8] wr_dqs_range = 4
[CortxA8] wr_data_range = 4
[CortxA8]
[CortxA8] Optimal values not reached, rerunning program with new values...
[CortxA8]
[CortxA8] ***************************************************************
[CortxA8]       The Slave Ratio Search Program Values are...
[CortxA8] ***************************************************************
[CortxA8] PARAMETER                       MAX  |  MIN  | OPTIMUM |  RANGE
[CortxA8] ***************************************************************
[CortxA8] DATA_PHY_RD_DQS_SLAVE_RATIO    0x06f | 0x005 |  0x03a  | 0x06a
[CortxA8] DATA_PHY_FIFO_WE_SLAVE_RATIO   0x1dd | 0x05a |  0x11b  | 0x183
[CortxA8] DATA_PHY_WR_DQS_SLAVE_RATIO    0x0cd | 0x05d |  0x095  | 0x070
[CortxA8] DATA_PHY_WR_DATA_SLAVE_RATIO   0x104 | 0x095 |  0x0cc  | 0x06f
[CortxA8] ***************************************************************
[CortxA8] rd_dqs_range = 0
[CortxA8] fifo_we_range = 1
[CortxA8] wr_dqs_range = 4
```

```
[CortxA8] wr_data_range = 4
[CortxA8]
[CortxA8] Optimal values not reached, rerunning program with new values...
[CortxA8]
[CortxA8] ****************************************************************
[CortxA8]       The Slave Ratio Search Program Values are...
[CortxA8] ****************************************************************
[CortxA8] PARAMETER                        MAX  |  MIN  | OPTIMUM |  RANGE
[CortxA8] ****************************************************************
[CortxA8] DATA_PHY_RD_DQS_SLAVE_RATIO     0x06f | 0x005 |  0x03a  | 0x06a
[CortxA8] DATA_PHY_FIFO_WE_SLAVE_RATIO    0x1df | 0x05b |  0x11d  | 0x184
[CortxA8] DATA_PHY_WR_DQS_SLAVE_RATIO     0x0d0 | 0x061 |  0x098  | 0x06f
[CortxA8] DATA_PHY_WR_DATA_SLAVE_RATIO    0x109 | 0x09a |  0x0d1  | 0x06f
[CortxA8] ****************************************************************
[CortxA8] rd_dqs_range = 0
[CortxA8] fifo_we_range = 2
[CortxA8] wr_dqs_range = 3
[CortxA8] wr_data_range = 5
[CortxA8]
[CortxA8] Optimal values not reached, rerunning program with new values...
[CortxA8]
[CortxA8] ****************************************************************
[CortxA8]       The Slave Ratio Search Program Values are...
[CortxA8] ****************************************************************
[CortxA8] PARAMETER                        MAX  |  MIN  | OPTIMUM |  RANGE
[CortxA8] ****************************************************************
[CortxA8] DATA_PHY_RD_DQS_SLAVE_RATIO     0x06f | 0x005 |  0x03a  | 0x06a
[CortxA8] DATA_PHY_FIFO_WE_SLAVE_RATIO    0x1dd | 0x05b |  0x11c  | 0x182
[CortxA8] DATA_PHY_WR_DQS_SLAVE_RATIO     0x0d3 | 0x064 |  0x09b  | 0x06f
[CortxA8] DATA_PHY_WR_DATA_SLAVE_RATIO    0x10a | 0x09c |  0x0d3  | 0x06e
[CortxA8] ****************************************************************
[CortxA8] rd_dqs_range = 0
[CortxA8] fifo_we_range = 1
[CortxA8] wr_dqs_range = 3
[CortxA8] wr_data_range = 2
[CortxA8]
[CortxA8] Optimal values have been found!!
[CortxA8]
[CortxA8] ===== END OF TEST =====
```

4. Once the program has finished running, use the 4 optimal values ((in the OPTIMUM column) to program the appropriate DDR PHY slave ratio register values. Also, use the values for CMD_PHY_INVERT_CLKOUT and CMD_PHY_CTRL_SLAVE_RATIO from the spreadsheet to program those registers. You can start by replacing the ones in the GEL and rerunning Scripts->AM335x System Initialization. Here is an example of where in the GEL to input the optimal values:

```
//*******************************************************************
//DDR3 PHY parameters
```

```
//*******************************************************************

#define  CMD_PHY_CTRL_SLAVE_RATIO       0x40
#define  CMD_PHY_INVERT_CLKOUT          0x1

#define  DATA_PHY_RD_DQS_SLAVE_RATIO    0x3A
#define  DATA_PHY_FIFO_WE_SLAVE_RATIO   0x11B //RD DQS GATE
#define  DATA_PHY_WR_DQS_SLAVE_RATIO    0xA8
#define  DATA_PHY_WR_DATA_SLAVE_RATIO   0xE1  //WRITE DATA
```

To test the results, open up a memory window in DDR space (0x80000000), and see if you can read/write values successfully. There are also a couple of DDR test scripts in the GEL which you can run. If the results look stable, then you have finished. These values should be programmed during your DDR initialization routine in your boot code (for example, in your linux bootloader). Be sure to program the same values for each iteration of CMDx registers (3 iterations) or DATAx registers (2 iterations). You can use these values for any board with the same layout and components.

5. Be sure to disable any HW leveling by setting REG_RDWRLVL_EN = 0 in the Read-Write Leveling Ramp Control Register in the EMIF.

## Procedural notes

Here are some notes on this procedure:

- If you rerun this program on the same board, you may get slightly different results. But these values could also be used. The program tries to find the proper timing window for reads and writes, and then chooses the middle of that window to allow for maximum timing margin. So these optimal values may be a little different with each run of this test program.
- The program defines the following ranges to determine when to stop looking for optimal values. These values were chosen based on multiple runs with different boards we tested.

1. define RD_DQS_OPTIMAL_RANGE 2
2. define WR_DQS_OPTIMAL_RANGE 3
3. define FIFO_WE_OPTIMAL_RANGE 2
4. define WR_DATA_OPTIMAL_RANGE 3

- If the program results in all zeros for values, then something has gone wrong and it was not able to converge on optimal values. Double check the values obtained in the spreadsheet and the values input while running the program in CCS.

## References

[1] http://processors.wiki.ti.com/index.php/AM335x_EMIF_Configuration_tips#DDR_PHY_Registers

[2] http://processors.wiki.ti.com/index.php/AM335x_EMIF_Configuration_tips#AC_timing_registers

# Article Sources and Contributors

**AM335x DDR PHY register configuration for DDR3 using Software Leveling** *Source*: http://processors.wiki.ti.com/index.php?oldid=158747 *Contributors*: Doublesin, Kevinsc

# Image Sources, Licenses and Contributors

**Image:RatioSeed.jpg** *Source*: http://processors.wiki.ti.com/index.php?title=File:RatioSeed.jpg *License*: unknown *Contributors*: Doublesin