Interfacing mDDR and DDR2 Memories with the i.MX51

by Multimedia Applications Division
Freescale Semiconductor, Inc.
Austin, TX

This application note describes the interfacing of Mobile Double Data Rate (mDDR) and Double Data Rate 2 (DDR2) memories with the i.MX51 processor. The application note also describes the routing guidelines with figures and examples.

1 i.MX51 SDRAM Controller

The i.MX51 Synchronous Dynamic Random Access Memory (SDRAM) controller can be interfaced with Single Data Rate SDRAM (SDR SDRAM), mDDR or DDR2 SDRAM memories. The DDR controller of the i.MX51 conforms to the immediate signals to interface with the memory devices.

The buses and signals associated with the SDRAM controller are as follows:

- Address bus and their bank control signals:
  - DRAM_A0—DRAM_A12
  - DRAM_SDBA0—DRAM_SDBA2

- Data bus and their buffer control signals:
  - DRAM_D0—DRAM_D31
- DRAM_DQS0/DQS0_B—DRAM_DQS3/DQS3_B
- DRAM_DQM0—DRAM_DQM3

- Control signals:
  - RAS
  - CAS
  - SDWE
  - DRAM_SDCKE0—DRAM_SDCKE1
  - DRAM_CS0—DRAM_CS1
  - DRAM_ODT0—DRAM_ODT1

- Clock signals:
  - DRAM_SDCLK
  - DRAM_SDCLK_B

Figure 1 shows the block diagram of the DDR2 interfaced with the i.MX51 using the Evaluation Kit (EVK) board.

![Diagram of interfacing i.MX51 and DDR2](image_url)

**Figure 1. Interfacing between i.MX51 and DDR2**
2 Memory Interfacing

This section describes the interfacing of the mDDR and DDR2 memories with the i.MX51 processor.

2.1 mDDR Interfacing

Figure 2 shows the interfacing between the i.MX51 and mDDR. The mDDR device used in Figure 2 is the MT46H64M16LFCK-5.

Figure 2. mDDR Memory Interfacing
2.2 DDR2 Interfacing

Figure 3 shows the interfacing between the i.MX51 and DDR2. The DDR2 device used in Figure 3 is the EDE1116AEBG.

The differences between the mDDR and DDR2 memories are as follows:

- The mDDR memories do not have the ODT and VREF signals, unlike the DDR2.
- The DQS signals are routed as differential pairs in DDR2 memories, unlike the mDDR.

**NOTE**

VREF is used as a reference voltage when ddr_input mode is enabled (non-CMOS mode) for DDR2 I/O. This mode is commonly used for DDR2 memories, especially if ODT (on-die termination) is enabled. For mDDR memories, typically CMOS mode is used and VREF is grounded. However, in some cases (due to board design, routing) non-CMOS mode may provide better performance. Therefore, making provision for VREF options allows flexibility.
3 DDR2 JTAG Script Configuration

The following code configures the DDR2 memory to interface with the i.MX51:

```c
//*==========================================================================================
//* Copyright (C) 2007, Freescale Semiconductor, Inc. All Rights Reserved
//* THIS SOURCE CODE IS CONFIDENTIAL AND PROPRIETARY AND MAY NOT
//* BE USED OR DISTRIBUTED WITHOUT THE WRITTEN PERMISSION OF
//* Freescale Semiconductor, Inc.
//*==========================================================================================
// Initialization script for Elvis EVB
// Revision History:
// Author (core ID) Date D/M/Y Number Description of Changes
// 26-Mar-2008 Taken from Marley, changed for
//==========================================================================================
wait = on

// init ARM
setreg @CP15_CONTROL=0x00050078
setreg @CPSR=0x1D3
// Setup PLL’s
setmem /32 0x83F80004 =0x00000000
setmem /32 0x83F84004 =0x00000000
setmem /32 0x83F88004 =0x00000000
setmem /32 0x83F80000 =0x00001222
setmem /32 0x83F80008 =0x00000070
setmem /32 0x83F8000C =23
setmem /32 0x83F88000 =0x00000222
setmem /32 0x83F88008 =0x00000091
setmem /32 0x83F8800C =0x00000000
setmem /32 0x83F88010 =0x00000000
// Switch ARM to PLL 3
setmem /32 0x73FD400C =0x00000180
setmem /32 0x73FD400C =0x00000184
setmem /32 0x83F80000 =0x00001232
pause 2
// Switch ARM back to PLL 1
setmem /32 0x73FD400C =0x000000180
setmem /32 0x73FD400C =0x000000184
// Restart PLL 3
setmem /32 0x83F80000 =0x00000232
pause 1
setmem /32 0x83F80004 =0x00000002
```
DDR2 JTAG Script Configuration

setmem /32 0x83F88004 =0x00000002

// Set the CCM registers
setmem /32 0x73fd4014 = 0x19239100
setmem /32 0x73fd4018 = 0x000020C2
setmem /32 0x73fd401C = 0xA5A2A020
setmem /32 0x73fd4024 = 0x00C30321
setmem /32 0x73fd4060 = 0x010900F0
setmem /32 0x73fa83F4 = 0x00000004
setmem /32 0x73fa83F0 = 0x00000004

// Initialization script for 32 bit DDR2 (CS0+CS1)
// Starting DDR frequency (after ROM code execution, before "prog_pll") is 150MHz
// Using validation environment (incl. "prog_pll" routine) will set DDR clock to 200MHz

// DDR2 IOMUX configuration
setmem /32 0x73fa88a0 = 0x00000200 //IOMUXC_SW_PAD_CTL_GRP_INMODE1 DDR2 mode
setmem /32 0x73fa850c = 0x000020c5 //IOMUXC_SW_PAD_CTL_PAD_EIM_SDODT1 Pull-D, HIGH DS
setmem /32 0x73fa8510 = 0x000020c5 //IOMUXC_SW_PAD_CTL_PAD_EIM_SDODT0 Pull-D, HIGH DS
setmem /32 0x73fa883c = 0x00000002 //IOMUXC_SW_PAD_CTL_GRP_DDR_A0 MEDIUM DS
setmem /32 0x73fa8848 = 0x00000002 //IOMUXC_SW_PAD_CTL_GRP_DDR_A1 MEDIUM DS
setmem /32 0x73fa84b8 = 0x0000000e7 //IOMUXC_SW_PAD_CTL_PAD_DRAM_SDCLK MAX DS
setmem /32 0x73fa84bc = 0x000000045 //IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS0 Pull/Keep Disable
setmem /32 0x73fa84c0 = 0x000000045 //IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS1 Pull/Keep Disable
setmem /32 0x73fa84c4 = 0x000000045 //IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS2 Pull/Keep Disable
setmem /32 0x73fa84c8 = 0x000000045 //IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS3 Pull/Keep Disable
setmem /32 0x73fa8820 = 0x000000000 //IOMUXC_SW_PAD_CTL_GRP_DDRPKS PUE = Keeper
setmem /32 0x73fa8444 = 0x00000003 //IOMUXC_SW_PAD_CTL_PAD_DRAM_RAS MEDIUM DS
setmem /32 0x73fa84a8 = 0x00000003 //IOMUXC_SW_PAD_CTL_PAD_DRAM_CAS MEDIUM DS
setmem /32 0x73fa84ac = 0x0000000e3 //IOMUXC_SW_PAD_CTL_PAD_DRAM_SDWE MEDIUM DS
setmem /32 0x73fa84bo = 0x0000000e3 //IOMUXC_SW_PAD_CTL_PAD_DRAM_SDCKE0 MEDIUM DS
setmem /32 0x73fa84b4 = 0x0000000e3 //IOMUXC_SW_PAD_CTL_PAD_DRAM_SDCKE1 MEDIUM DS
setmem /32 0x73fa84cc = 0x0000000e3 //IOMUXC_SW_PAD_CTL_PAD_DRAM_CS0 MEDIUM DS
setmem /32 0x73fa84cd = 0x0000000e2 //IOMUXC_SW_PAD_CTL_PAD_DRAM_CS1 MEDIUM DS

setmem /32 0x73FA882C = 0x4 // DRAM_D31-24 to high
setmem /32 0x73FA88a4 = 0x4 // DRAM_D0-7 to high
setmem /32 0x73FA88ac = 0x4 // DRAM_D8-15 to high
setmem /32 0x73FA88b6 = 0x4 // DRAM_D16-23 to high

// 13 ROW, 10 COL, 32Bit, SREF=4 Micron Model
// CAS=3, BL=4
setmem /32 0x83fd9000 = 0x82a20000 // ESDECTL0
setmem /32 0x83fd9008 = 0x82a20000 // ESDECTL1
setmem /32 0x83fd9010 = 0x0000ad0d0 // ESDMISC
setmem /32 0x83fd9004 = 0x3F3584AB // ESDECFG0
setmem /32 0x83fd900c = 0x3F3584AB // ESDECFG1

// Init DRAM on CS0
setmem /32 0x83fd9014 = 0x04008008 // ESDESCR
setmem /32 0x83fd9014 = 0x0000801a // ESDESCR
setmem /32 0x83fd9014 = 0x0000801b // ESDESCR
setmem /32 0x83fd9014 = 0x00448019 // ESDESCR
4 mDDR and DDR2 Routing Rules

The routing is one of the critical procedures while interfacing. The guidelines for the routing are restrictive, and the difficulty to meet the routing guidelines depend on the stackup. However, the routing helps the design to become better and more robust.

The DDR routing is done in two ways:

- Routing all the signals at the same length
- Routing by byte-group

The two routing methods are described as follows:

- Routing all the signals at the same length is a difficult method. However, it is the better way, as it helps easy analysis.
Table 1 gives the specifications for routing the signals at the same length.

Table 1. Specifications for Routing Signals at the Same Length

<table>
<thead>
<tr>
<th>Signals</th>
<th>Lengths</th>
<th>Considerations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address and Bank</td>
<td>≤ Clock length</td>
<td>Match the signals ±25 mils</td>
</tr>
<tr>
<td>Data and Buffer</td>
<td>≤ Clock length</td>
<td></td>
</tr>
<tr>
<td>Control signals</td>
<td>≤ Clock length</td>
<td></td>
</tr>
<tr>
<td>Clock</td>
<td>≤ Lcritical (3 inches)</td>
<td>Match the signals with the clock signals ±5 mils</td>
</tr>
<tr>
<td>DQS and DQS_B</td>
<td>≤ Clock length</td>
<td>Match the signals with the DQS signals ±10 mils</td>
</tr>
</tbody>
</table>

- Routing by byte-group requires a better control of the signals of each group and is more difficult for analysis and constraints settings.

Table 2 gives the specifications for routing by byte-group.

Table 2. Specifications for Routing by Byte-Group

<table>
<thead>
<tr>
<th>Signals</th>
<th>Length</th>
<th>Considerations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address and Bank</td>
<td>(Clock length - 200 mils) ≤ Address and Bank length ≤ Clock length</td>
<td>Match the signals ±25 mils</td>
</tr>
<tr>
<td>Byte group 1</td>
<td>DQ0—DQ7, DQS0, DQM0 The max byte group 1 length ≤ Clock length</td>
<td>Match the signals of each byte group ±25 mils</td>
</tr>
<tr>
<td>Byte group 2</td>
<td>DQ8—DQ15, DQS1, DQM1 The max byte group 2 length ≤ Clock length</td>
<td>The difference between the byte-groups should be ±50 mils. Match the signals with the DQS signals ±10 mils. (Clock length - 500 mils) ≤ DQS Signals Length ≤ Clock length</td>
</tr>
<tr>
<td>Byte group 3</td>
<td>DQ16—DQ23, DQS2, DQM2 The max byte group 3 length ≤ Clock length</td>
<td></td>
</tr>
<tr>
<td>Byte group 4</td>
<td>DQ24—DQ31, DQS3, DQM3 The max byte group 4 length ≤ Clock length</td>
<td></td>
</tr>
<tr>
<td>Control signals</td>
<td>(Clock length - 200 mils) ≤ Control length ≤ Clock length</td>
<td>Match the signals ±50 mils</td>
</tr>
<tr>
<td>Clock</td>
<td>≤ Lcritical (3 inches)</td>
<td>Match the signals with the clock signals ± 5 mils</td>
</tr>
</tbody>
</table>

NOTE

The impedance for the signals should be 50 Ω for singled ended and 100 Ω for differential pairs.

5 Routing on i.MX51 EVK

This section describes the EVK implementation for routing the DDR2 memories.
The Figure 4 shows the placement of the i.MX51 in the EVK board. This procedure should be followed to have the memories close to the i.MX51 and for a better control of the signal length.

**Figure 4. DDR2 Memories and i.MX51 Placement**

Figure 5 and Figure 6 show the different ways to route the DDR signals in point to point connection. This type of connection reduces the number of vias and allows routing without layer transitions.

**Figure 5. Point to Point Connection**

**Figure 6. Point to Point Connection Including the Series Resistor**
The stackups used for this type of design are as follows:

- 8 layers
  - L1—top signal
  - L2—GND plane
  - L3—internal signal 1
  - L4—PWR plane
  - L5—PWR plane
  - L6—internal signal 2
  - L7—GND plane
  - L8—bottom signal

- 6 layers
  - L1—top signal/power
  - L2—GND plane
  - L3—internal signal 1/power
  - L4—internal signal 2/power
  - L5—GND plane
  - L6—bottom signal/power

Figure 7, Figure 8, Figure 9, Figure 10, and Figure 11, give an example for routing DDR2 memories. This example in effect is the guidelines for layer routing.

- Figure 7 shows the top DDR2 routing.
- Figure 8 shows the internal 1 DDR2 routing.

![Figure 8. Internal 1 DDR2 Routing](image)

- Figure 9 shows the power 1 DDR2 routing.

![Figure 9. Power 1 DDR2 Routing](image)
• **Figure 10** shows the internal 2 DDR2 routing.

![Figure 10. Internal 2 DDR2 Routing](image)

• **Figure 11** shows the bottom DDR2 routing.

![Figure 11. Bottom DDR2 Routing](image)
Table 3 gives the total etch of the signals for the byte 0 and byte 1 groups.

### Table 3. Etch for DDR2 Routing

<table>
<thead>
<tr>
<th>Signals</th>
<th>Length (mils)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRAM_D0</td>
<td>1141.69</td>
</tr>
<tr>
<td>DRAM_D1</td>
<td>1151.25</td>
</tr>
<tr>
<td>DRAM_D2</td>
<td>1156.71</td>
</tr>
<tr>
<td>DRAM_D3</td>
<td>1057.88</td>
</tr>
<tr>
<td>DRAM_D4</td>
<td>1155.42</td>
</tr>
<tr>
<td>DRAM_D5</td>
<td>1152.43</td>
</tr>
<tr>
<td>DRAM_D6</td>
<td>1159.87</td>
</tr>
<tr>
<td>DRAM_D7</td>
<td>1182.69</td>
</tr>
<tr>
<td>DRAM_DQM0</td>
<td>1138.23</td>
</tr>
<tr>
<td>DRAM_SDQS0</td>
<td>1156.36</td>
</tr>
<tr>
<td>DRAM_SDQS0_B</td>
<td>1192.52</td>
</tr>
<tr>
<td>DRAM_D8</td>
<td>1037.69</td>
</tr>
<tr>
<td>DRAM_D9</td>
<td>1039.47</td>
</tr>
<tr>
<td>DRAM_D10</td>
<td>1046.71</td>
</tr>
<tr>
<td>DRAM_D11</td>
<td>1038.47</td>
</tr>
<tr>
<td>DRAM_D12</td>
<td>1039.58</td>
</tr>
<tr>
<td>DRAM_D13</td>
<td>1132.83</td>
</tr>
<tr>
<td>DRAM_D14</td>
<td>1057.36</td>
</tr>
<tr>
<td>DRAM_D15</td>
<td>1056.28</td>
</tr>
<tr>
<td>DRAM_DQM1</td>
<td>10.41.13</td>
</tr>
<tr>
<td>DRAM_SDQS1</td>
<td>1046.25</td>
</tr>
<tr>
<td>DRAM_SDQS1_B</td>
<td>1045.87</td>
</tr>
<tr>
<td>DRAM_SDCLK</td>
<td>1249.02</td>
</tr>
<tr>
<td>DRAM_SDCLK_B</td>
<td>1295.91</td>
</tr>
</tbody>
</table>

6 Guidelines for 13 x 13 mm Package

The 13 x 13 mm package has 0.5 mm pitch. So, it is necessary to use the via across the pad and different types of vias, such as buried, blind or stacked.

The design considerations are given as follows:

- Blind via—10 mils pad and 4 mils drill from the top to the internal layer
- Ball Grid Array (BGA) spacing—3.34 mils gap, 3 mils width, and 3.34 mils gap
- Trace width for DDR routing—3 mils
• Stackup for the 10 layers are given as follows:
  — L1—top signal
  — L2—GND plane
  — L3—internal signal 1
  — L4—GND plane
  — L5—power plane
  — L6—power plane
  — L7—GND plane
  — L8—internal signal 2
  — L9—GND plane
  — L10—bottom signal

The strategy to route the mDDR interface are given as follows:
• Use the top side to route in the horizontal way
• Only the two columns of pins, near to the edge of the IC should be routed on the top
• The internal signal 1 and internal signal 2 should be used for DDR routing in horizontal ways and vertical ways, respectively
• Bottom layer should be used for vertical memory connections

Figure 12, Figure 13, Figure 14, and Figure 15 give an example for routing mDDR memories. This example in effect is the guidelines for layer routing.
• Figure 12 shows top DDR routing.
• Figure 13 shows top internal signal 1 routing.

Figure 13. Internal Signal 1 Routing

• Figure 14 shows internal signal 2 routing.

Figure 14. Internal Signal 2 Routing
Figure 15 shows bottom DDR routing.

![Figure 15. Bottom DDR Routing](image)

Table 4 shows the total etch of the signals for the byte 0 and byte 1 groups.

<table>
<thead>
<tr>
<th>Signals</th>
<th>Length (mils)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRAM_D0</td>
<td>1170.21</td>
</tr>
<tr>
<td>DRAM_D1</td>
<td>1151.90</td>
</tr>
<tr>
<td>DRAM_D2</td>
<td>1152.96</td>
</tr>
<tr>
<td>DRAM_D3</td>
<td>1161.40</td>
</tr>
<tr>
<td>DRAM_D4</td>
<td>1149.45</td>
</tr>
<tr>
<td>DRAM_D5</td>
<td>1151.89</td>
</tr>
<tr>
<td>DRAM_D6</td>
<td>1168.71</td>
</tr>
<tr>
<td>DRAM_D7</td>
<td>1150.21</td>
</tr>
<tr>
<td>DRAM_SDQS0</td>
<td>1150.74</td>
</tr>
<tr>
<td>DRAM_DQM0</td>
<td>1150.05</td>
</tr>
<tr>
<td>DRAM_D8</td>
<td>1168.12</td>
</tr>
<tr>
<td>DRAM_D9</td>
<td>1152.29</td>
</tr>
</tbody>
</table>
Table 4. Etch for Byte 0 and Byte 1 Groups (continued)

<table>
<thead>
<tr>
<th>Signals</th>
<th>Length (mils)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRAM_D10</td>
<td>1149.49</td>
</tr>
<tr>
<td>DRAM_D11</td>
<td>1153.24</td>
</tr>
<tr>
<td>DRAM_D12</td>
<td>1149.34</td>
</tr>
<tr>
<td>DRAM_D13</td>
<td>1151.75</td>
</tr>
<tr>
<td>DRAM_D14</td>
<td>1156.11</td>
</tr>
<tr>
<td>DRAM_D15</td>
<td>1150.63</td>
</tr>
<tr>
<td>DRAM_SDQS1</td>
<td>1166.04</td>
</tr>
<tr>
<td>DRAM_DQM1</td>
<td>1171.87</td>
</tr>
<tr>
<td>DRAM_SDCLK</td>
<td>2217.78</td>
</tr>
<tr>
<td>DRAM_SDCLK_B</td>
<td>2190.78</td>
</tr>
</tbody>
</table>

7 Revision History

Table 5 provides a revision history for this application note.

Table 5. Document Revision History

<table>
<thead>
<tr>
<th>Rev. Number</th>
<th>Date</th>
<th>Substantive Change(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>10/2010</td>
<td>Added a note in Section 2.2, “DDR2 Interfacing.”</td>
</tr>
<tr>
<td>1</td>
<td>06/2010</td>
<td>In Table 2, changed the clock length:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• 200 to 200 mils</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• 500 to 500 mils</td>
</tr>
<tr>
<td>0</td>
<td>04/2010</td>
<td>Initial release.</td>
</tr>
</tbody>
</table>
Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters which may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including “Typicals” must be validated for each customer application by customer’s technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

Freescale, the Freescale logo, CodeWarrior, ColdFire, PowerQUICC, StarCore, and Symphony are trademarks of Freescale Semiconductor, Inc. Reg. U.S. Pat. & Tm. Off. CoreNet, QorIQ, QUICC Engine, and VortiQa are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. ARM is the registered trademark of ARM Limited.

© 2010 Freescale Semiconductor, Inc.