## Contents

1 Revision History ......................................................... 1  
  1.1 Revision 4.0 ......................................................... 1  
  1.2 Revision 3.0 ......................................................... 1  
  1.3 Revision 2.0 ......................................................... 1  
  1.4 Revision 1.0 ......................................................... 1  

2 Error Detection and Correction on SmartFusion2 Devices using DDR Memory . . . . 2  
  2.1 Introduction ......................................................... 2  
  2.2 Design Requirements ............................................... 3  
  2.3 Demo Design ........................................................ 3  
    2.3.1 Demo Design Implementation .................................. 4  
    2.3.2 Loop Test ....................................................... 4  
    2.3.3 Manual Test ..................................................... 5  
  2.4 Setting Up the Demo Design ....................................... 6  
    2.4.1 Graphical User Interface ...................................... 7  
  2.5 Running the Demo Design .......................................... 8  
    2.5.1 Performing Loop Test ........................................... 9  
    2.5.2 Performing Manual Test ....................................... 9  
  2.6 Conclusion ......................................................... 10  

3 Appendix: Jumper Settings .............................................. 11
## Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Top-Level Block Diagram</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>Demo Design Top-Level Structure</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>Design Flow</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>USB to UART Bridge Drivers</td>
<td>6</td>
</tr>
<tr>
<td>5</td>
<td>SmartFusion2 Advanced Development Kit Board Setup</td>
<td>7</td>
</tr>
<tr>
<td>6</td>
<td>DDR – EDAC Demo GUI</td>
<td>7</td>
</tr>
<tr>
<td>7</td>
<td>FlashPro Programming Window</td>
<td>8</td>
</tr>
<tr>
<td>8</td>
<td>Initialization Completed Window</td>
<td>9</td>
</tr>
<tr>
<td>9</td>
<td>1-bit Error Loop Detection Window</td>
<td>10</td>
</tr>
<tr>
<td>10</td>
<td>2-bit Error Detection Manual Window</td>
<td>10</td>
</tr>
</tbody>
</table>
Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 1</td>
<td>Design Requirements</td>
<td>3</td>
</tr>
<tr>
<td>Table 2</td>
<td>DDR3 Memory Addresses used in Loop Test</td>
<td>9</td>
</tr>
<tr>
<td>Table 3</td>
<td>DDR3 Memory Addresses used in Manual Test</td>
<td>9</td>
</tr>
<tr>
<td>Table 4</td>
<td>SmartFusion2 Advanced Development Kit Jumper Settings</td>
<td>11</td>
</tr>
</tbody>
</table>
1 Revision History

The revision history describes the changes that were implemented in the document. The changes are listed by revision, starting with the most current publication.

1.1 Revision 4.0
Updated the document for Libero v11.8 software release.

1.2 Revision 3.0
Updated the document for Libero v11.7 software release.

1.3 Revision 2.0
Updated the document for Libero v11.6 software release.

1.4 Revision 1.0
Initial release for Libero SoC v11.5 software release.
2 Error Detection and Correction on SmartFusion2 Devices using DDR Memory

2.1 Introduction

In a single event upset (SEU) susceptible environment, random access memory (RAM) is prone to transient errors caused by heavy ions.

This document describes the EDAC capabilities of the SoC FPGA, which are used in applications with memories connected through the microcontroller subsystem (MSS) DDR (MDDR).

The EDAC controllers implemented in the SmartFusion2 devices support single error correction and double error detection (SECDED). All memories—enhanced static random access memory (eSRAM), DDR, low-power DDR (LPDDR)—within the SmartFusion2 MSS devices are protected by SECDED. The DDR synchronous dynamic random access memory (SDRAM) can be DDR2, DDR3, or LPDDR1, depending on the MDDR configuration and hardware ECC capabilities.

The SmartFusion2 MDDR subsystem supports memory densities up to 4 GB. In this demo, you can select any memory location of 1 GB in the DDR address space (0xA0000000 to 0xDFFFFFFF).

When SECDED is enabled:

- A write operation computes and adds 8 bits of SECDED code (to every 64 bits of data)
- A read operation reads and checks the data against the stored SECDED code to support 1-bit error correction and 2-bit error detection

The following illustration describes the block diagram of SmartFusion2 EDAC on DDR SDRAM.

*Figure 1* • Top-Level Block Diagram

The EDAC feature of DDR supports the following:

1. SECDED mechanism
2. Provides interrupts to the ARM Cortex-M3 processor and FPGA fabric upon the detection of a 1-bit error or 2-bit error
3. Stores the number of 1-bit and 2-bit errors in error counter registers
4. Stores the address of the last 1-bit or 2-bit error affected memory location
5. Stores the 1-bit or 2-bit error data in SECDED registers
6. Provides error bus signals to the FPGA fabric

For more information about EDAC, see UG0443: SmartFusion2 and IGLOO2 FPGA Security and Reliability User Guide and UG0446: SmartFusion2 and IGLOO2 FPGA High-Speed DDR Interfaces User Guide.
2.2 Design Requirements

The following table lists the design requirements.

<table>
<thead>
<tr>
<th>Design Requirements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hardware Requirements</strong></td>
<td></td>
</tr>
<tr>
<td>SmartFusion2 Advanced Development Kit board</td>
<td>Rev B or later</td>
</tr>
<tr>
<td>FlashPro5 programmer or later</td>
<td></td>
</tr>
<tr>
<td>USB A to mini-B USB cable</td>
<td></td>
</tr>
<tr>
<td>Power adapter</td>
<td>12 V</td>
</tr>
<tr>
<td>DDR3 Daughter board</td>
<td></td>
</tr>
<tr>
<td>Operating System</td>
<td>Any 64-bit or 32-bit Windows XP SP2</td>
</tr>
<tr>
<td>Any 64-bit or 32-bit Windows 7</td>
<td></td>
</tr>
<tr>
<td><strong>Software Requirements</strong></td>
<td></td>
</tr>
<tr>
<td>Libero® System-on-Chip (SoC)</td>
<td>v11.8</td>
</tr>
<tr>
<td>SoftConsole</td>
<td>v4.0</td>
</tr>
<tr>
<td>FlashPro programming software</td>
<td>v11.8</td>
</tr>
<tr>
<td>Host PC Drivers</td>
<td>USB to UART drivers</td>
</tr>
<tr>
<td>Framework to run demonstration</td>
<td>Microsoft .NET Framework 4 client</td>
</tr>
</tbody>
</table>

2.3 Demo Design

The demo design files are available for downloading from the following path in the Microsemi website: http://soc.microsemi.com/download/rsc/?f=m2s_dg0618_liberov11p8_df

The demo design files include:
- DDR Configuration File
- DDR_EDAC
- Programming files
- GUI executable
- Readme file
The following illustration describes the top-level structure of the design files. For further details, see the readme.txt file.

**Figure 2 • Demo Design Top-Level Structure**

![Diagram](<download_folder>)

### 2.3.1 Demo Design Implementation

The MDDR subsystem has a dedicated EDAC controller. EDAC detects a 1-bit error or 2-bit error when data is read from the memory. If EDAC detects the 1-bit error, the EDAC controller corrects the error bit. If EDAC is enabled for all the 1-bit and 2-bit errors, corresponding error counters in the system registers are incremented and corresponding interrupts and error bus signals to the FPGA fabric are generated.

This happens in real-time. To demonstrate this SECDED feature, an error is introduced manually and observed detection and correction.

This demo design involves implementation of following steps:

1. Enable EDAC
2. Write data to DDR
3. Read data from DDR
4. Disable EDAC
5. Corrupt 1 or 2 bits
6. Write data to DDR
7. Enable EDAC
8. Read the data
9. In the case of a 1-bit error, the EDAC controller corrects the error, updates the corresponding status registers, and gives the data written in Step 2 at the read operation done at Step 8.
10. In the case of a 2-bit error, a corresponding interrupt is generated and the application must correct the data or take the appropriate action in the interrupt handler. These two methods are demonstrated in this demo.

Two tests are implemented in this demo: loop test and manual test and they are applicable to both 1-bit and 2-bit errors.

### 2.3.2 Loop Test

Loop test is executed when the SmartFusion2 devices receive a loop test command from the GUI. Initially, all the error counters and EDAC related registers are placed in the RESET state.

The following steps are executed for each iteration.

1. Enable the EDAC controller
2. Write the data to the specific DDR memory location
3. Disable the EDAC controller
4. Write the 1-bit or 2-bit error induced data to the same DDR memory location
5. Enable the EDAC controller
6. Read the data from the same DDR memory location
7. Send the 1-bit or 2-bit error detection and 1-bit error correction data in case of 1-bit error to the GUI
### 2.3.3 Manual Test

This method allows manual testing of 1-bit error detection and correction and 2-bit error detection for DDR memory address (0xA0000000 to 0xDFFFFFFF) with initialization. A 1-bit/2-bit error is introduced manually to a selected DDR memory address. The given data is written to the selected DDR memory location with EDAC enabled. The corrupted 1-bit or 2-bit error data is then written to the same memory location with EDAC disabled. The information on the detected 1-bit or 2-bit error is logged when the data is read from the same memory location with EDAC enabled. The high-performance DMA controller (HPDMA) is used to read the data from the DDR memory. The dual-bit error detection interrupt handler is implemented to take the appropriate action when a 2-bit error is detected.

The following illustration describes the EDAC demo operations.

**Note:** For a 2-bit error, when the Cortex-M3 processor reads the data, the code execution goes in to the hard fault handler, as the interrupt received is late for the processor to respond. By the time it responds to the interrupt, it may have already passed the data and accidentally launched a command. As a result, the HRESP stops processing the incorrect data. 2-bit error detection uses HPDMA to read the data from the DDR address location, which instructs the processor that read data has a 2-bit error and the system should take appropriate action to recover (ECC interrupt Handler).
2.4 Setting Up the Demo Design

This section describes the SmartFusion2 Advanced Development Kit board setup, the GUI options, and how to execute the demo design.

The following steps describe how to setup the demo:

1. Connect one end of the USB mini-B cable to the **J33** connector provided in the SmartFusion2 Advanced Development Kit board. Connect the other end of the USB cable to the host PC. Light emitting diode (LED) DS27 must light up, indicating the UART link has been established. Ensure that the USB to UART bridge drivers are automatically detected (can be verified in the **Device Manager**), as shown in the following figure.

   **Figure 4 • USB to UART Bridge Drivers**

   ![USB to UART Bridge Drivers](image)

   If USB to UART bridge drivers are not installed, download and install the drivers from: [www.microsemi.com/soc/documents/CDM_2.08.24_WHQL_Certified.zip](http://www.microsemi.com/soc/documents/CDM_2.08.24_WHQL_Certified.zip).

2. Connect the jumpers on the SmartFusion2 Advanced Development Kit board, as shown in **Table 4**, page 11. The power supply switch **SW7** must be switched OFF, while making the jumper connections.
2.4.1 Graphical User Interface

This section describes the DDR – EDAC Demo GUI.

The GUI supports the following features:
1. Selection of COM port and Baud Rate
2. Selection of 1-bit error correction tab or 2-bit error detection
3. Address field to write or read data to or from specified DDR address
4. Data field to write or read data to or from specified DDR address
5. Serial Console section to print the status information received from the application
6. Enable EDAC/Disable EDAC: Enables or disables the EDAC
7. Write: Allows writing data to the specified address
8. **Read**: Allows reading data from the specified address
9. **Loop test ON/OFF**: Allows testing the EDAC mechanism in a loop method
10. **Initialize**: Allows to initialize the predefined memory location (in this demo A0000000-A000CFFF)

### 2.5 Running the Demo Design

The following steps describe how to run the design:

1. Switch ON the supply switch, **SW7**.
2. Program the SmartFusion2 device with the programming file provided in the design files. ([\ProgrammingFile\EDAC_DDR3.stp](#)) using FlashPro design software, as shown in the following figure.

![FlashPro Programming Window](image)

3. Press **SW6** switch to reset the board after successful programming.
4. Launch the **EDAC_DDR Demo GUI** executable file available in the design files ([\GUI Executable\EDAC_DDR.exe](#)). The GUI window is displayed, as shown in Figure 8, page 9.
5. Click **Connect**, it selects the COM port and establishes the connection. **Connect** option changes to **Disconnect**.
6. Select the **1-bit Error Correction** tab or **2-bit Error Detection**.
7. Manual and Loop tests can be performed.
8. Click **Initialize** to initialize the DDR memory to perform the Manual and Loop tests, an **initialization completion** message is displayed on Serial Console, as shown in Figure 8, page 9.
2.5.1 Performing Loop Test

Click **Loop Test ON**. It runs in loop mode where continuous correction and detection of errors is done. All actions performed in the SmartFusion2 device are logged in the **Serial Console** section of the GUI.

<table>
<thead>
<tr>
<th>Table 2</th>
<th>DDR Memory Addresses used in Loop Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory</td>
<td>DDR3</td>
</tr>
<tr>
<td>1-bit error correction</td>
<td>0xA0008000</td>
</tr>
<tr>
<td>2-bit error detection</td>
<td>0xA000C000</td>
</tr>
</tbody>
</table>

2.5.2 Performing Manual Test

In this method, errors are introduced manually using GUI. Use the following steps to execute 1-bit error correction or 2-bit error detection.

<table>
<thead>
<tr>
<th>Table 3</th>
<th>DDR Memory Addresses used in Manual Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory</td>
<td>DDR3</td>
</tr>
<tr>
<td>1-bit error correction</td>
<td>0xA0000000-0xA0004000</td>
</tr>
<tr>
<td>2-bit error detection</td>
<td>0xA0004000-0xA0008000</td>
</tr>
</tbody>
</table>

Input Address and Data fields (use 32-bit Hexadecimal values).

1. Click **Enable EDAC**.
2. Click **Write**.
3. Click **Disable EDAC**.
4. Change one bit (in case of 1-bit error correction) or two bits (in case of 2-bit error detection) in **Data** field (introducing error).
5. Click **Write**.
6. Click **Enable EDAC**.
7. Click **Read**.
8. Observe Error Count Display and Data field in the GUI. The error count value increases by 1.
2.6 Conclusion

This demo shows SmartFusion2 SECDED capabilities for the MDDR subsystem.
Appendix: Jumper Settings

The following table shows all the required jumpers to set on the SmartFusion2 Advanced Development Kit.

Table 4 • SmartFusion2 Advanced Development Kit Jumper Settings

<table>
<thead>
<tr>
<th>Jumper</th>
<th>Pin (From)</th>
<th>Pin (To)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>J116, J353, J354, J54</td>
<td>1</td>
<td>2</td>
<td>These are the default jumper settings of the Advanced Development Kit board. Ensure these jumpers are set accordingly.</td>
</tr>
<tr>
<td>J123</td>
<td>2</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>J124, J121, J32</td>
<td>1</td>
<td>2</td>
<td>JTAG programming via FTDI</td>
</tr>
</tbody>
</table>